НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра системного програмування і
спеціалізованих комп'ютерних систем

«До захисту допущено»

Завідувач кафедри

_____ Тарасенко В.П.

«___»_____2019 р.

Дипломний проект

на здобуття ступеня бакалавра

з напряму підготовки 6.050102 «Комп'ютерна інженерія»

на тему: «Система автоматизації розрахунку харчування на борту»

Виконав:
студент IV курсу, групи КВ-52
Довжик Дмитро Вікторович _____

Керівник:
Доцент кафедри СПіСКС, к.т.н., доцент
Сапсай Т.Г. _____

Консультант з нормоконтролю:
Доцент кафедри СПіСКС, к.т.н., доцент
Клятченко Я.М. _____

Рецензент:

_____

Засвідчую, що у цьому дипломному проекті немає запозичень з праць інших авторів без відповідних посилань.
Студент _____

Київ – 2019 року

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет прикладної математики

Кафедра системного програмування і спеціалізованих комп‘ютерних систем

Рівень вищої освіти – перший (бакалаврський)

Напрям підготовки 6.050102 «Комп'ютерна інженерія»

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Тарасенко В.П.
(підпис) (ініціали, прізвище)

«___» червня 2019 р.

ЗАВДАННЯ
на дипломний проект студента
Довжика Дмитра Вікторовича

1. Тема проекту «Система автоматизації розрахунку харчування на борту». Керівник проекту Сапсай Тетяна Григорівна, доц. каф. СПіСКС, к.т.н., затверджені наказом по університету від 22 травня 2019 р. №1330-С

2. Термін подання студентом проекту 12 червня 2019 р.

3. Вихідні дані до проекту: див. технічне завдання.

4. Зміст пояснювальної записки: аналіз існуючих рішень та постановка задачі, обґрунтування вибору засобів реалізації системи, розробка системи автоматизації розрахунку харчування на борту, опис роботи програмного продукту.

5. Перелік графічного матеріалу (із зазначенням обов’язкових креслеників, плакатів, презентацій тощо): архітектура системи, організація інтерфейсу взаємодії з користувачем, формування замовлення для кейтерингової компанії, авторизація на веб-сервері.

6. Консультанти розділів проекту*

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Клятченко Я.М., доц. каф. СПіСКС, к.т.н.		

7. Дата видачі завдання 17.12.2018

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1.	Видача завдання на дипломне проектування	17.12.2018	
2.	Розроблення та узгодження технічного завдання	11.02.2019	
3.	Аналіз існуючих рішень	25.02.2019	
4.	Формалізація постановки задачі	22.04.2019	
5.	Вибір засобів реалізації системи	29.04.2019	
6.	Розробка та відлагодження системи	08.05.2019	
7.	Підготовка пояснювальної записки	15.05.2019	
8.	Підготовка графічної частини дипломного проекту	23.05.2019	
9.	Оформлення документації дипломного проекту	27.05.2019	
10.	Попередній огляд матеріалів дипломного проекту на кафедрі	29.05.2019	

Студент _____ Довжик Д.В.
(підпис)

Керівник проекту _____ Сапсай Т.Г.
(підпис)

* Консультантом не може бути зазначено керівника дипломного проекту.

АНОТАЦІЯ

Дипломний проект включає пояснювальну записку (62 ст., 32 рис., 8 табл.).

Об’єктом розробки є система розрахунку харчування на борту літака.

Метою даного проекту є розробка системи автоматизації, яка забезпечує формування замовлень для кейтерингової компанії, що обслуговує екіпаж та пасажирів на борту повітряного судна.

В ході виконання дипломного проекту було:

- проведено аналіз існуючих рішень;
- розроблено алгоритм формування замовлення для кейтерингової компанії;
- розроблено систему розрахунку харчування на борту повітряного судна;
- представлено роботу системи.

Використання даної системи забезпечує автоматизацію формування замовлення для кейтерингової компанії, що обслуговує екіпаж та пасажирів на борту повітряного судна.

Ключові слова:

автоматизація, алгоритм, система, веб-сервер, авторизація, HTTP, C++, Qt, фреймворк, QML, GUI, MVC.

ABSTRACT

The diploma project includes explanatory note (62 pages, 32 drawings, 8 tables).

The object of development is the system for automating the calculation of nutrition on board an aircraft.

The purpose of this project is development of automation system, which provides for the formation of orders for a catering company serving the crew and passengers on board an aircraft.

During the development of the diploma project:

- analyzed existing solutions;
- developed the algorithm of order formation for the catering company;
- developed the system for calculating nutrition on board an aircraft;
- presented the work of the system.

Using this system provides automation for forming of orders for a catering company serving the crew and passengers on board an aircraft.

Keywords:

automation, algorithm, system, web server, authorization, HTTP, C++, Qt, framework, QML, GUI, MVC.

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
			Документація загальна			
			Новорозроблена			
	А4	ІАЛЦ.045490.002 ТЗ	Система автоматизації	4		
			розрахунку харчування			
			на борту.			
			Технічне завдання			
	А4	ІАЛЦ.045490.003 ТП	Система автоматизації	2		
			розрахунку харчування			
			на борту.			
			Відомість технічного			
			проекту			
	А4	ІАЛЦ.045490.004 ПЗ	Система автоматизації	62		
			розрахунку харчування			
			на борту.			
			Пояснювальна записка			
	А4	ІАЛЦ.045490.005 Д1	Архітектура системи.	1		
			Схема структурна			
	А4	ІАЛЦ.045490.006 Д2	Організація інтерфейсу	1		
			взаємодії з користувачем.			
			Схема структурна			

Змін.	Арк.	№ докум.	Підпис	Дата	ІАЛЦ.045490.001 ОА			
Розробив		Довжик Д.В.			Система автоматизації розрахунку харчування на борту.	Літ.	Аркуш	Аркушів
Перевірив		Сапсай Т.Г.					1	2
						НТУУ "КПІ ім. Ігоря Сікорського", ФПМ, КВ-52		
Н. контроль		Клятченко Я.М.			Опис альбому			
Затвердив		Тарасенко В.П.						

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
	А4	ІАЛЦ.045490.007 Д3	Формуванння замовлення	1		
			для кейтерингової			
			компанії.			
			Схема алгоритму			
	А4	ІАЛЦ.045490.008 Д4	Авторизація на веб-сервері.	1		
			Схема алгоритму			
		Диск CD-ROM	Текст ПЗ. Тексти програм.	1		
			Графічні матеріали			

Зміст

1. НАЙМЕНУВАННЯ І ОБЛАСТЬ ЗАСТОСУВАННЯ _____2
2. ПІДСТАВА ДЛЯ РОЗРОБКИ _____2
3. ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ_____2
4. ДЖЕРЕЛА РОБОТИ _____2
5. ТЕХНІЧНІ ВИМОГИ _____2
   5.1. Вимоги до апаратного забезпечення _____2
   5.2. Вимоги до програмного забезпечення_____3
   5.3. Вимоги до системи, що розробляється _____3
6. ВИМОГИ ДО ПРОЕКТНОЇ ДОКУМЕНТАЦІЇ _____3
7. ЕТАПИ РОЗРОБКИ _____4

1. НАЙМЕНУВАННЯ І ОБЛАСТЬ ЗАСТОСУВАННЯ

Найменування роботи – «Система автоматизації розрахунку харчування на борту».

Область застосування: автоматизація процесу розробки меню для пасажирів та екіпажу на борту повітряного судна.

2. ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки є завдання на виконання дипломного проекту першого (бакалаврського) рівня вищої освіти, затверджене кафедрою системного програмування та спеціалізованих комп’ютерних систем Національного технічного університету України «Київський Політехнічний Інститут імені Ігоря Сікорського».

3. ЦІЛЬ І ПРИЗНАЧЕННЯ РОБОТИ

Метою даного проекту є розробка системи автоматизації, яка забезпечує формування замовлень для кейтерингової компанії, що обслуговує літак та пасажирів на борту повітряного судна.

4. ДЖЕРЕЛА РОБОТИ

Джерелами інформації є науково-технічна література, публікації у періодичних виданнях та Інтернет ресурси з питань розробки та автоматизації процесу створення та розрахунку меню для громадського харчування.

5. ТЕХНІЧНІ ВИМОГИ

5.1. Вимоги до апаратного забезпечення:

- комп'ютер на базі процесора Intel Pentium;
- оперативна пам’ять – 4 Гбайт і більше;
- наявність доступу до мережі Інтернет.

5.2. Вимоги до програмного забезпечення:

- операційна система Windows;
- 64-розрядна версія ОС;
- вільне місце на диску – 300 Мбайт.

5.3. Вимоги до системи, що розробляється:

- підтримка взаємодії з користувачем через графічний інтерфейс;
- наявність інтерфейсу для авторизації працівника авіакомпанії;
- формування замовлення у вигляді таблиці в форматі .xlsx;
- надання інформації користувачу в разі зміни необхідних даних на веб-сервері;
- наявність інтерфейсу для розрахунку ваги бортприпасів і обладнання.

6. ВИМОГИ ДО ПРОЕКТНОЇ ДОКУМЕНТАЦІЇ

У процесі виконання проекту повинна бути розроблена наступна документація:

- пояснювальна записка;
- креслення:
  - «Архітектура системи. Схема структурна»;
  - «Організація інтерфейсу взаємодії з користувачем. Схема структурна»;
  - «Формування замовлення для кейтерингової компанії. Схема алгоритму»;
  - «Авторизація на веб-сервері. Схема алгоритму».

7. ЕТАПИ РОЗРОБКИ

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів
1.	Видача завдання на дипломне проектування	17.12.2018
2.	Розроблення та узгодження технічного завдання	11.02.2019
3.	Аналіз існуючих рішень	25.02.2019
4.	Формалізація постановки задачі	22.04.2019
5.	Вибір засобів реалізації системи	29.04.2019
6.	Розробка та відлагодження системи	08.05.2019
7.	Підготовка пояснювальної записки	15.05.2019
8.	Підготовка графічної частини дипломного проекту	23.05.2019
9.	Оформлення документації дипломного проекту	27.05.2019
10.	Попередній огляд матеріалів дипломного проекту на кафедрі	29.05.2019

Поз.	Формат	ПОЗНАЧЕННЯ	НАЙМЕНУВАННЯ	Кількість аркушів	№ прим.	Примітки
			Документація загальна			
			Новорозроблена			
	А4	ІАЛЦ.045490.004 ПЗ	Система автоматизації	62		
			розрахунку харчування			
			на борту.			
			Пояснювальна записка			
	А4	ІАЛЦ.045490.005 Д1	Архітектура системи.	1		
			Схема структурна			
	А4	ІАЛЦ.045490.006 Д2	Організація інтерфейсу	1		
			взаємодії з користувачем.			
			Схема структурна			
	А4	ІАЛЦ.045490.007 Д3	Формуванння замовлення	1		
			для кейтерингової			
			компанії.			
			Схема алгоритму			
	А4	ІАЛЦ.045490.008 Д4	Авторизація на веб-сервері.	1		
			Схема алгоритму			
		Диск CD-ROM	Текст ПЗ. Тексти програм.	1		
			Графічні матеріали			

Змін.	Арк.	№ докум.	Підпис	Дата	ІАЛЦ.045490.003 ТП			
Розробив		Довжик Д.В.			Система автоматизації розрахунку харчування на борту.	Літ.	Аркуш	Аркушів
Перевірив		Сапсай Т.Г.					1	1
Консульт.						НТУУ "КПІ ім. Ігоря Сікорського", ФПМ, КВ-52		
Н. контроль		Клятченко Я.М.			Відомість технічного проекту			
Зав. каф.		Тарасенко В.П.						

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ 3

ВСТУП 6

1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАДАЧІ 8

1.1 Організація харчування на борту повітряного судна 8

1.1.1 Особливості харчування на повітряних суднах 8

1.1.2 Види харчування на повітряних суднах 10

1.1.3 Харчування для екіпажу 10

1.2 Існуючі рішення систем розрахунку харчування на борту 10

1.3 Формалізація постановки задачі 11

1.4 Обґрунтування теми дипломного проекту 19

2. ВИБІР ТЕХНОЛОГІЇ РОЗРОБКИ СИСТЕМИ З ГРАФІЧНИМ ІНТЕРФЕЙСОМ 20

2.1 Обґрунтування вибору технології розробки системи 20

2.2 Бібліотека Qt 27

3. РОЗРОБКА СИСТЕМИ АВТОМАТИЗАЦІЇ РОЗРАХУНКУ ХАРЧУВАННЯ НА БОРТУ 31

3.1 Загальна структура системи 31

3.2 Головний модуль 34

3.3 Модуль зв'язку з сервером 34

3.4 Модуль відстеження змін даних 39

3.5 Модуль формування замовлення 41

3.6 Модуль обрахунку ваги бортприпасів і обладнання 47

3.7 Графічний інтерфейс 48

3.8 Опис роботи системи 49

ВИСНОВКИ..... 59

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ..... 61

ДОДАТКИ

Додаток 1. Копії графічних матеріалів

ІАЛЦ.045490.005 Д1. Архітектура системи. Схема структурна.

ІАЛЦ. 045490.006 Д2. Організація інтерфейсу взаємодії з користувачем. Схема структурна.

ІАЛЦ. 045490.007 Д3. Формування замовлення для кейтерингової компанії. Схема алгоритму.

ІАЛЦ. 045490.008 Д4. Авторизація на веб-сервері. Схема алгоритму.

Додаток 2. Лістинг програмного коду

Додаток 3. Акт про впровадження результатів дипломного проекту

Додаток 4. Презентація

ПЕРЕЛІК СКОРОЧЕНЬ, УМОВНИХ ПОЗНАЧЕНЬ, ТЕРМІНІВ

Віджет (Widget)	— графічний елемент або модуль інтерфейсу.
Віртуальна машина	— модель обчислювальної машини, яка створена шляхом віртуалізації обчислювальних ресурсів: процесора, оперативної пам'яті, пристроїв зберігання та вводу і виводу інформації.
Графічний інтерфейс користувача (Graphical user interface, GUI)	— тип інтерфейсу, який дозволяє користувачам взаємодіяти з електронними пристроями через графічні зображення та візуальні вказівки.
Кейтерингова компанія	— компанія, що надає послуги з приготування і доставки продуктів, організації процесу харчування.
Клієнт	— персональний комп'ютер або робоча станція, що запрошує інформацію у сервера.
Прикладний програмний інтерфейс (Application Programming Interface, API)	— інтерфейс взаємодії, що є набором класів, структур, функцій, констант, що представлений додатком або операційною системою для використання у зовнішніх програмних продуктах.
ПС	— повітряне судно.
Сервер	— комп'ютер (як правило, високопродуктивна робоча станція, міні-комп'ютер), що зберігає інформацію, з якою працюють клієнти.

Фреймворк	— заготовки, шаблони для програмної платформи, що визначають архітектуру програмної системи.
BF Group	— назва компанії, яка поставляє безмитні товари на продаж на борт повітряного судна.
HTTP (HyperText Transfer Protocol)	— прикладний протокол передачі даних в мережі, використовується для отримання інформації з веб-сайтів.
HTTPS (HyperText Transfer Protocol Secure)	— розширення протоколу HTTP, яке підтримує шифрування даних, використовуючи криптографічний протокол TLS.
Microsoft .NET Framework	— платформа, що створена компанією Microsoft, для розробки та виконання клієнтський та серверних програм і розрахована на роботу під операційними системами сімейства Microsoft Windows.
PHP (PHP: Hypertext Preprocessor)	— мова програмування, спеціально розроблена для написання веб-додатків, що виконуються на веб-сервері.
Sky Food	— вид меню, яке пропонується для пасажирів повітряного судна.
TLS (Transport Layer Security)	— тип цифрової безпеки, який дозволяє зашифрувати зв'язок між веб-сервером та веб-клієнтом.
WINDROSE (Роза вітрів)	— українська авіакомпанія, що здійснює чартерні та регулярні авіарейси, і є замовником системи автоматизації

	розрахунку харчування на борту, яка розробляється у даному дипломному проекті.
Y/C	— позначення кількості людей на борту повітряного судна.

ВСТУП

Якісне харчування в літаку є однією з складових конкурентних особливостей сучасного авіаційного транспорту. Даний вид послуг залежить від часу, проведеного в польоті, і класу обслуговування, обраного пасажиром при придбанні квитка. Кожна, як українська, так і зарубіжна авіакомпанія пропонує різні раціони в залежності від класу, яким летить пасажир, від його віку, кулінарних та релігійних уподобань, медичних показань і, навіть, від моделі літака (у деяких літаках не передбачена можливість розігріти їжу).

Харчування на борту повітряного судна має свою специфіку, наприклад, певні терміни і умови зберігання продуктів, спеціальний спосіб комплектації, послуга доставки харчування на борт, зняття з літака і обробка використаного обладнання та посуду і т.д. Все це потребує наявності спеціальної техніки та розроблених технологічних процесів.

Організацією процесу харчування, як правило, займається кейтерингова компанія. Функції та перелік послуг, які може надавати така компанія при організації бортового харчування, значно відрізняються від тих, що пропонуються традиційними закладами ресторанного типу, за способами організації, обсягами робіт і тривалістю виконання замовлень. Функції кейтерингової компанії полягають в наступному: прийняття замовлень на харчування від представників авіакомпаній; приготування страв згідно із замовленням; забезпечення певних умов завантаження повітряного судна та особливостей обслуговування пасажирів в польоті.

Проведений аналіз особливостей організації харчування на борту дозволив визначити основні проблеми, які виникають у представників кейтерингової компанії, що забезпечує процес підготовки та приготування страв у відповідності до вимог авіакомпанії.

Для оптимізації роботи працівників авіакомпанії, що відповідають за організацію харчування на борту, необхідно створити систему

автоматизації розрахунку харчування на борту, яка забезпечить взаємодію між відповідними працівниками авіакомпанії та кейтерингової компанії.

Метою даного проекту є розробка системи автоматизації, яка забезпечує формування замовлень для кейтерингової компанії, що обслуговує екіпаж та пасажирів на борту повітряного судна.

Запропонована система дозволяє:

1) формувати "Лист замовлення" для кейтерингової компанії та здійснювати необхідні для цього обрахунки, на основі розкладу рейсів літаків, списків екіпажу та інформації на веб-сайті авіакомпанії про кількість і замовлення пасажирів;

2) відслідковувати зміну номеру рейсу, маршруту, дату і час вильоту, кількості пасажирів та екіпажу на кожному з рейсів, кількості та видів замовлених меню;

3) розраховувати вагу бортприпасів і обладнання на кожному з рейсів повітряного судна в залежності від заданих параметрів, пов'язаних з організацією процесу харчування на борту ПС.

1. АНАЛІЗ ІСНУЮЧИХ РІШЕНЬ ТА ПОСТАНОВКА ЗАДАЧІ

1.1 Організація харчування на борту повітряного судна

1.1.1 Особливості харчування на повітряних суднах

Харчування є одним із найважливіших зовнішніх факторів, які впливають на здоров'я людини. Останнім часом якісне харчування під час польоту стає однією з основних вимог пасажирів до авіакомпаній. Харчування на повітряних суднах класифікують за певними ознаками: раціоном харчування, контингентом споживачів. За раціоном харчування виділяють: сніданки, обіди, обіди-вечері, а також звичайний та дієтичний раціон. За контингентом споживачів розрізняють: харчування для пасажирів та екіпажу. За замовником харчування поділяється на харчування, яке замовляється авіакомпанією, та харчування, що надається пасажирам (екіпажу) регулярних і чартерних рейсів.

Прийняття замовлення на харчування від представників авіакомпаній здійснюється за день до вильоту повітряного судна з аеропорту, враховуючи вимоги авіакомпанії за кількістю та асортиментом харчування для пасажирів різних класів, дієт, особистих побажань. У цехах кейтерингового підприємтва, що знаходяться в аеропорту, готують сніданки, обіди і вечері для споживання в дорозі. Цехи складаються з двох відділень: для розфасування продуктів та укладання їх у контейнери. У фасувальне відділення з холодного цеху надходять закуски, із гарячого – бульйон та другі страви. Перед укладкою в контейнери холодні закуски і другі страви порціонують. Готову їжу доставляють в літак у спеціальних контейнерах. На підприємстві щоденно складається виробнича програма згідно із замовленням, розкладом польотів літаків та затвердженим асортиментним переліком страв, що видаються на борт. Специфіка кейтерингового підприємства полягає в тому, що виробництво продукції здійснюється чітко за розкладом авіарейсів [1]. Сировина, яка використовується на підприємстві, спочатку надходить у складські

приміщення, де зберігається, а потім передається у заготівельні цехи - овочевий та м'ясо-рибний і проходить обробку. Технологічний процес виробництва кулінарної продукції можна поділити на декілька стадій:

1) механічна кулінарна обробка продуктів, яка включає операції з приготування напівфабрикатів, нарізання та порціонування овочів, фруктів;

2) теплова кулінарна обробка продуктів, при якій напівфабрикати доводяться до готовності, після чого у спеціальній холодильній шафі швидко охолоджуються до температури +5°С;

3) порціонування, фасування та упакування страв на пакувальній лінії, на якій операції здійснюються вручну та із застосуванням конвеєра;

4) транспортування страв на борт повітряних суден для подальшої реалізації пасажирам.

Після приготування холодних страв та закусок їх порціонують і транспортують до відділу фасування. Для короткочасного зберігання сировини, напівфабрикатів та готової продукції передбачена холодильна камера. При цьому враховується, що повна готовність страв і кулінарних виробів має бути за дві години до запланованого авіарейсу.

Для порціонування готових страв використовується різний посуд - пластиковий, фарфоровий, металевий, скло, залежно від рівня авіакомпанії. Харчування для кожної авіакомпанії фасується у фірмовий посуд (з логотипом авіакомпанії) та на касалетки, які відрізняються за кольором та формою. Крім того, розробляється і попередньо затверджується асортимент продукції, закупних товарів та порядок розміщення посуду зі стравами, столових наборів, серветок, пакетиків цукру, чаю тощо на касалетці. Для спрощення роботи кухарів, фасувальників, підтримання якості оформлення та подавання страв розробляються карти-схеми з чіткими вказівками стосовно асортименту страв, переліку і розміщення їх та іншого інвентарю. Кожну розфасовану

касалетку закривають харчовою плівкою та складають у контейнери, на які наклеюється стікер, де вказуються строки реалізації продукції і відповідальний за їх завантаження [2].

1.1.2 Види харчування на повітряних суднах

Більшість українських авіакомпаній надають чотири основних види харчування в літаку: стандартне, вегетаріанське, кошерне, спеціальне, яке готується з урахуванням вимог клієнта.

Спеціальне харчування - особливий вид страв, які не включені ні в одне з основних видів меню. Для того щоб замовити спеціальне харчування в літаку, слід надати замовлення за дві доби до вильоту у авіакомпанії. При собі потрібно мати квиток на конкретний рейс, а також документи, що підтверджують особистість. Даний вид харчування користується попитом у пасажирів, яким за станом здоров'я потребують особливого режиму харчування [3].

1.1.3 Харчування для екіпажу

До складу екіпажу входять командир, другий пілот, штурман та бортпровідники. Залежно від типу повітряного судна команда може бути і більшою. Раціон харчування екіпажу відрізняється від страв, які пропонуються пасажирам. Для командира і другого пілота передбачений різний набір продуктів, для уникнення загрози отруєння обох пілотів. Час і порядок прийому їжі членами екіпажу встановлюється старшим бортпровідником екіпажу.

1.2 Існуючі рішення систем розрахунку харчування на борту

Існуючі системи для розрахунку забезпечення харчування пасажирів і екіпажу повітряного судна інтегровані у системи авіакомпаній. Наприклад, кейтерингова авіакомпанія LSG Sky Chef є частиною компанії

авіаперевізника Lufthansa. Система розрахунку харчування компанії LSG Sky Chef інтегрована до системи Lufthansa і використовує інформацію щодо рейсів та кількості пасажирів з системи авіаперевізника. Дана система не може бути використаною для іншої авіакомпанії. Основна мета дипломного проекту полягає в тому, щоб розробити систему автоматизації розрахунку харчування на борту повітряного судна, яка буде інтегрованою до системи авіакомпанії WINDROSE.

1.3 Формалізація постановки задачі

Для розробки системи автоматизації розрахунку харчування на борту, потрібно проаналізувати роботу працівника відділу організації харчування авіакомпанії, який забезпечує зв'язок з кейтерінговою компанією. Завдання працівника: сформувати "Лист замовлення" для підприємства, яке займається приготуванням і транспортуванням їжі та наданням додаткових послуг. "Лист замовлення" — це стандартна форма замовлення бортового харчування для екіпажу, пасажирів та додаткових послуг (побутового обладнання, товарів для торгівлі на борту літака) (див. рис. 1.3.1).

Для заповнення інформації про рейс літака, працівник спочатку отримує на пошту розклад рейсів літаків на 3 дні з поточної дати (див. рис 1.3.2). Інформація про дату і номер рейсу, маршрут рейсу та тип ПС переноситься у "Лист замовлення".

Інформація про кількість екіпажу на борту надсилається для працівника поштою (див. рис. 1.3.3). Харчування для командира ПС та 2-го пілота замовляється окремо від інструктора і бортпровідників. Інструктор — спеціаліст, який є членом екіпажу, що забезпечує тренування членів екіпажу з метою допуску їх до виконання польотів на ПС даної авіакомпанії. Меню для інструктора розраховується так само, як для бортпровідників.

WINDROSE

Диспетчер-координатор:

Дата	№ рейсу	Тип ПС/ Бортовий	Направлення, час UTC	Пілоти		Екіпаж		Sky Food		Бізнес/Преміум/Sky menu		харчування		Бортприпаси, норма	BF group	Побутове обладнання
				Туди	Зворотнє	Туди	Зворотнє	Туди	Зворотнє	Туди	Зворотнє	Туди	Зворотнє			
07.03.2019	7W 7653/ 7W 6654	A-321/UR-WRJ	KBP 01:10-SSH 05:15/06:15-LWO 10:20 Y/C - напої + платне харчування	1\1	1\1	5	5 + розрахунок бортприпасів на 14 Crew	7W41Y21 - 2 пор 7W41Y19 - 2 пор **ТК 24 год** 7W11Y37 - 2 пор 7W41Y21 - 2 пор		*	*	*	*	**4-5 год** (без харчування) Розрахунок напоїв та посуду з КВР на всі перельоти 872 Y/C	**Sky-Shop Sky Food**	**Побутове на 4 перельоти (2 комплект) для Єгипту**
07.03.2019	7W 6653/ 7W 7654	A-321/UR-WRJ	LWO 11:40-SSH 15:45/16:45-KBP 20:50 **Y/C - напої з КВР 400 г лимону (нарізаного порціями по 100 г), 2 кг льоду Рівноцінний обмін обладнання**	2	2	5	5	Замовлено з КВР		*	*	*	*	**Бортприпаси з КВР**	**BF group з КВР**	**Побутове обладнання з КВР**
07.03.2019	7W 7051/ 7W 3052	A-321/UR-WRT	KBP 01:30-SSH 05:35/06:35-DNK 10:40 Y/C -напої + платне харчування	1\1	1\1	5	5+ розрахунок бортприпасів на14 Crew	7W41Y21 - 2 пор 7W41Y19 - 2 пор **ТК 24 год** 7W11Y37 - 2 пор 7W41Y21 - 2 пор		*	1 ELMP 1 FLMP 1 MPSK 1 MSSK 2 MTSK 1 SFSK (для 3052) --- 1 ELMP 4 CHML 1 LCML 1 MOML 1 MPSK 1 MRSK (для 3051 во ТК 17	*	*	**4-5 год** (без харчування) Розрахунок бортприпасів з КВР на всі перельоти 800 Y/C	**Sky-Shop Sky Food**	**Побутове на 4 перельоти (2 комплекти) для Єгипту**
07.03.2019	7W 3051/ 7W 7052	A-321/UR-WRT	DNK 12:00-SSH 16:05/17:05-KBP 21:10 **Y/C - напої з КВР 400 г лимону (нарізаного порціями по 100 г), 2 кг льоду Рівноцінний обмін обладнання**	2	2	5	5	Замовлено з КВР		**1 ELMP 4 CHML 1 LCML 1 MOML 1 MPSK 1 MRSK**	*	*	*	**Бортприпаси з КВР**	**BF group з КВР**	**Побутове з КВР**

Рис. 1.3.1 Приклад "Листа замовлення" від авіакомпанії

Оновлений розклад рейсів з 07 по 09 березня 2019 року

Planning <planning@windrose.kiev.ua>

Отправлено: Ср 06.03.2019 13:26

Кому:

Час UTC:

Примітки	Дата	Рейс	Маршрут	Тип ПС/Борт
07.03.19 Четвер				
	07.03.19	7W 7653/7W 6654	KBP 01:10-SSH 05:15/06:15-LWO 10:20	A-321/UR-WRJ
	07.03.19	7W 7051/7W 3052	KBP 01:30-SSH 05:35/06:35-DNK 10:40	A-321/UR-WRT
	07.03.19	7W 7663/7W 7664	KBP 02:05-HRG 06:00/07:00-KBP 11:00	A-321/UR-WRV
	07.03.19	7W 6653/7W 7654	LWO 11:40-SSH 15:45/16:45-KBP 20:50	A-321/UR-WRJ
	07.03.19	7W 3051/7W 7052	DNK 12:00-SSH 16:05/17:05-KBP 21:10	A-321/UR-WRT
	07.03.19	7W 7965/7W 7966	KBP 12:10-HRG 16:05/17:05-KBP 21:05	A-321/UR-WRV
	07.03.19	7W 106/7W 105	DNK 07:00-KBP 08:10/16:00-DNK 17:10	E145/UR-DNR
	07.03.19	7W 755/7W 756	KBP 07:05-SOF 09:20/10:00-KBP 12:00	E145/UR-DNG
	07.03.19	7W 101/7W 102	KBP 07:40-DNK 08:50/10:00-KBP	E145/UR-DNF

Рис. 1.3.2. Приклад розкладу рейсів на 3 дні

07.03.2019 Четвер	Борт №	Рейс №	Маршрут	Вилiт UTC	Екiпаж				Понадштатні члени
					Командир	2 пiлот	Інструктор	Бортпровiдники	
	UR-WRJ	7W7653 7W6654	KBP - SSH SSH - LWO	01:10-05:15 06:15-10:20	Рябой М.В LTC [0/75]	Біпоусько А.С 2 пилот		Стах В.Л ІБПВ Тюхніна О.О БП Усаченко Л.В БП Гринюк І.М БП Ярмак О.М БП	
	UR-WRJ	7W6653 7W7654	LWO - SSH SSH - KBP	11:40-15:45 16:45-20:50	Бабій В.В КПС [0/75]	Грайцар Р.С 2 пилот		Федоріна Г.І СБП Карбовська Е.М БП Лепеха К.В БП Лесик А.П БП Мартинюк В.О БП	

Рис. 1.3.3. Приклад списку членів екіпажу

Бортове харчування поділено на 3 групи:

1) "Sky Food";

2) "Бізнес/Преміум/Sky menu";

3) "Харчування Y/C".

Працівник самостійно заповняє стовпчики "Sky Food" та "BF Group" згідно норми для даного рейсу.

Інформація про харчування "Бізнес/Преміум/Sky menu" та Харчування Y/C береться із веб-сайту авіакомпанії, у який заносяться всі замовлення спеціального харчування або додаткових сервісів пасажирів літака. На рисунку 1.3.4 відображено вигляд веб-сайта для працівника відділу організації харчування авіакомпанії, де надається необхідна інформація про рейси.

Рис. 1.3.4. Загальний вигляд веб-сайта для працівника

На рисунку 1.3.5, для прикладу, відображено інформацію про один із рейсів авіакомпанії.

Flights 07.03.2019

№	Serv	Flight	Route	Date	Dep	Seats			Pax				Meal	
						C	W	Y	C	W	Y		Code	Count
8		7W 3052	SSH-DNK	07.03.2019	08:35			220	**0**	**0**	**180 + 6(24)**	**180 + 6(24)**	ELMP	1
													FLMP	1
													MPSK	1
													MSSK	1
													MTSK	2
													SFSK	1
													WCHC	1
													WCHS	1
													Y	156
													Y Chd	24
													Y Inf	6

Рис. 1.3.5. Інформація про рейс

В графі "Meal" відображені спеціальні замовлення, які були зроблені пасажирами рейсу — ELMP, FLMP, MPSK, MSSK, MTSK, SFSK, WCHC, WCHS — це коди, що позначають позиції "Sky Menu" та додаткові сервіси, що не відносяться до харчування пасажирів і не враховуються працівником відділу організації харчування. Для прикладу, код ELMP, який має транскрипцію — "Обід/Яловичина тушкована з кус-кусом" і відноситься до харчування пасажирів та код WCHS — "пасажир потребує інвалідний візок", що не відноситься до харчування. В таблиці 1.3.1 наведено коди з даного прикладу, що позначають позиції "Sky Menu", і їх транскрипції.

Таблиця 1.3.1

Приклад транскрипцій кодів позицій "Sky Menu"

Код харчування	Транскрипція
ELMP	FSM Обід/ Яловичина тушкована з кус-кусом
FLMP	FSM Обід/ Стейк із індички та пюре із броколі
MPSK	FSM Обід/ Куряча брізоль та паста
MSSK	FSM Обід/ Чахохбілі з картопляним пюре
MTSK	FSM Обід/ Бургер з пастою
SFSK	Сендвіч «Міні-хлібець а-ля «Цезар»

Дані коди з графи "Meal" для даного рейсу переносяться у стовпчик "Бізнес/Преміум/Sky menu" із таблиці "Лист замовлення".

Різниця між графою "Туди" та "Зворотнє" полягає в тому що, графа "Зворотнє" заповнюється для рейсів, які з закордону прямують в українські аеропорти (див. табл. 1.3.2). Відповідно графа "Туди" заповнюється для усіх інших напрямів.

Таблиця 1.3.2

Аеропорти України, що обслуговуються авіакомпанією "WINDROSE"

Позначення	Назва аеропорту
KBP	Міжнародний аеропорт «Бориспіль»
DNK	Міжнародний аеропорт «Дніпропетровськ»
LWO	Міжнародний аеропорт «Львів» імені Данила Галицького
OZH	Міжнародний аеропорт «Запоріжжя»
ODS	Міжнародний аеропорт «Одеса»
HRK	Міжнародний аеропорт «Харків»

Графа "Бортприпаси, норма" відображає норму напоїв для пасажирів та екіпажу. Надання прохолодних напоїв є обов'язковою безкоштовною послугою на борту повітряного судна. На кожну годину польоту мінімальна норма надання прохолодних напоїв становить 100 мл, а в разі відсутності харчування - 150 мл на годину польоту. У період з 15 квітня до 14 жовтня включно норма надання прохолодних напоїв збільшується до 150 мл. Тому для замовлення напоїв (та відповідного посуду), у замовленні вказується загальна кількість пасажирів та екіпажу на усіх рейсах даного напряму.

Так як авіакомпанія і підприємство, яке надає послуги з приготуванню харчування, знаходяться в аеропорту "Бориспіль" (KBP) в Києві, то для правильного завантаження бортприпасів аналізується розклад рейсів (приклад на рис. 1.3.2). Нехай, літак вилітає з Києва в місто іншої країни, наприклад Шарм-еш-Шейх (SSH, країна Єгипет), а потім звідти в будь-яке українське місто, наприклад, Львів (LWO) — це два рейси першого напряму — KBP-SSH-LWO. Два інші рейси другого напряму — це повернення літака через Шарм-еш-Шейх до Києва — LWO-SSH-KBP. Тоді ці два напрями складають один цикл повітряного судна і вони згруповані в таблиці "Лист замовлення". Кількість бортприпасів вказується для першого та другого напряму і записується: "Бортприпаси з KBP", як це показано у таблиці 1.3.3.

Таблиця 1.3.3

Приклад циклу польотів одного літака, відображеного, в таблиці "Лист замовлення"

Дата	**№ рейсу**	**Тип ПС/ Бортовий**	**Направлення, час UTC**	**Бортприпаси, норма**	**Побутове обладнання**
07.03.2019	7W 7653/ 7W 6654	A-321/ UR-WRJ	KBP 01:10 - SSH 05:15/06:15 - LWO 10:20	870 Y/C + 14 C	Побутове обладнання на 4 перельоти (2 комплекти)
07.03.2019	7W 6653/ 7W 7654	A-321/UR-WRJ	LWO 11:40 - SSH 15:45/16:45 - KBP 20:50	Бортприпаси з KBP	Побутове обладнання з KBP

Далі для підрахунку кількості бортприпасів, з веб-сайту береться інформація по кожному з рейсів цих двох напрямів (див. рис. 1.3.6), де в графі "Meal" в рядку "Y" вказана кількість дорослих пасажирів, а в рядку "Y chd" вказана кількість дітей.

Flights 07.03.2019

№	Serv	Flight	Route	Date	Dep	Seats			Pax					Meal	
						C	W	Y	C		W		Y	Code	Count
1		7W 7653	KBP-SSH	07.03.2019	03:10			218	**0**		**0**		**218 + 5(32)**	**218 + 5(32)**	Y	186
															Y Chd	32
															Y Inf	5
7		7W 6654	SSH-LWO	07.03.2019	08:15			218	**0**		**0**		**216 + 9(40)**	**216 + 9(40)**	Y	176
															Y Chd	40
															Y Inf	9
19		7W 6653	LWO-SSH	07.03.2019	14:15			218	**0**		**0**		**218 + 4(43)**	**218 + 4(43)**	Y	175
															Y Chd	43
															Y Inf	4
26		7W 7654	SSH-KBP	07.03.2019	18:45			218	**0**		**0**		**218 + 2(20)**	**218 + 2(20)**	Y	198
															Y Chd	20
															Y Inf	2

Рис. 1.3.6. Приклад інформації на веб-сайті про рейси двох напрямів, які складають цикл польоту повітряного судна

Для даного прикладу кількість пасажирів дорівнює:

7W7653 + 7W6654 + 7W6653 + 7W7654 = (186 + 32) + (176 + 40) + (175 + 43) + (198 + 20) = 218 + 216 + 218 + 218 = 870.

Також додається кількість членів екіпажу, яка береться з списку членів екіпажу по кожному з напрямів (див. рис. 1.3.3):

7W7653|7W6654 + 7W6653|7W7654 = 7 + 7 = 14.

Стовпчик "Побутове обладнання" заповнюється в залежності від кількості перельотів (рейсів) на напрямах: "Побутове обладнання на 2 перельоти (розділити на 1 переліт)", "Побутове обладнання на 2 перельоти", "Побутове обладнання на 4 перельоти".

При цьому, якщо є два напрями, які створюють цикл, то для першого вказується один із варіантів комплекту, а для другого: "Побутове обладнання з KBP".

Для того, щоб оптимізувати роботу працівника авіакомпанії, що відповідає за організацію харчування на борту повітряного судна, потрібно створити систему, яка автоматизує розрахунок харчування для пасажирів та екіпажу рейсів літаків, тобто дозволить:

1) формувати "Лист замовлення" для кейтерингової компанії та здійснювати необхідні для цього обрахунки, на основі розкладу рейсів літаків, списків екіпажу та інформації на веб-сайті авіакомпанії про кількість і замовлення пасажирів;

2) відслідковувати зміну номеру рейсу, маршруту, дату і час вильоту, кількості пасажирів та екіпажу на кожному з рейсів, кількості та видах замовлених меню;

3) окремо мати інтерфейс для обрахунку ваги літака на кожному з рейсів повітряного судна в залежності від заданих параметрів, пов'язаних з організацією харчування на борту ПС.

1.4 Обґрунтування теми дипломного проекту

Створення системи, яка автоматизує розрахунок харчування людей на борту повітряного судна дозволить зробити більш ефективнішою роботу відділу авіакомпанії, що відповідає за організацію харчування на борту, спростить введення і пошук потрібної в роботі інформації.

Виходячи з того, що в компанії авіаперевізника WINDROSE система, яка використовується працівниками компанії, що відповідають за організацію харчування на борту ПС, не забезпечує можливість повного автоматизованого формування документів для кейтерингової компанії, розробка системи автоматизації розрахунку харчування на борту є актуальною. Створення такої системи буде простим і зручним інструментом для користувачів, дозволить забезпечити чітку взаємодію між відповідними працівниками авіакомпанії та кейтерингової компанії.

2. ВИБІР ТЕХНОЛОГІЇ РОЗРОБКИ СИСТЕМИ З ГРАФІЧНИМ ІНТЕРФЕЙСОМ

При виборі технології для розробки програмного проекту необхідно врахувати безліч різних аспектів, найбільш важливим з яких є мова програмування, тому що вона в значній мірі визначає інші доступні засоби. Наприклад, для розробки графічного інтерфейсу розробникам необхідна GUI-бібліотека, що надає готові елементи інтерфейсу, такі, як кнопки і меню. Так як вибір GUI-бібліотеки дуже впливає на розробку проекту, часто її вибір здійснюється першим, а мова програмування визначається з числа доступних для цієї бібліотеки мов. Зазвичай, мова програмування визначається бібліотекою однозначно [4].

Основними технологіями для розробки графічного інтерфейсу являються мова програмування C++ з бібліотекою Qt, Java з бібліотеками AWT/Swing та C# з платформою Microsoft .NET Framework. У цьому розділі розглянуто саме ці технології.

2.1 Обґрунтування вибору технології розробки системи

Продуктивність програмування

Продуктивність програмування визначає, наскільки ефективно (тобто швидко і точно) програміст з певним досвідом і знаннями може вирішити поставлену перед ним задачу, використовуючи задану мову програмування.

Особливістю мов Java і C# в порівнянні з іншими мовами програмування загального призначення є забезпечення високої продуктивності програмування, ніж швидкість роботи програми або ефективність використання ними пам'яті.

Для цього мови програмування Java та C# наділені деякими додатковими можливостями. Наприклад, на відміну від мови C ++ (або C), програміст не повинен в явному вигляді "звільняти" (повертати) виділену

пам'ять операційній системі. Звільнення невикористаної пам'яті ("збирання сміття") автоматично забезпечується середовищем виконання мовами Java чи C#, що знижує продуктивність і ефективність використання пам'яті. Це звільняє програміста від додаткового завдання: стеження за звільненням пам'яті — головного джерела помилок в програмах. Ця можливість мов Java та C# повинна значно збільшити продуктивність процесу програмування в порівнянні з мовою C ++ (або C) [5].

Програми, що написані мовою програмування Java чи C# повільніші, аніж ті, що написані мовою C++. Пояснюється це тим, що програми мовою C++ компілюються компілятором у двійковий формат, який потім виконується безпосередньо процесором. Таким чином, виконання програми здійснюється апаратними засобами. З іншої сторони, компілятор мови Java компілює вихідний код в "байт-код", який безпосередньо виконується не процесором, а за допомогою іншого програмного забезпечення, віртуальної машини Java (Java Virtual Machine, JVM). В свою чергу віртуальна машина JVM виконується процесором. Таким чином, виконання байт-коду програм мовою Java виконується не швидкими апаратними засобами, а за допомогою більш повільної програмної емуляції. У випадку з мовою C#, код виконується у "загальномовному середовищі виконання" Common Language Runtime (CLR) і має такий же принцип роботи як віртуальна машина JVM, тому швидкість виконання програм мовою C# менша, аніж у програм, написаних мовою C++.

Ефективність використання пам'яті

Мови програмування Java і C ++ використовують різні підходи в управлінні пам'яттю. У мові C++ управління пам'яттю повністю здійснюється програмістом, тобто в міру необхідності розподіл і звільнення пам'яті повинно виконуватися програмістом. Мова Java забезпечує автоматичне звільнення невикористовуваної пам'яті. Поряд з розподілом пам'яті програмістом, віртуальна машина JVM веде облік всіх

використовуваних блоків пам'яті і вказівників на них. Якщо блок пам'яті більше не використовується, він може бути звільнений. Це забезпечує процес, який називається "збирання сміття". Він періодично викликається віртуальною машиною JVM, перевіряє всі використовувані блоки пам'яті і звільняє ті з них, на які відсутні вказівники.

"Збирання сміття" зручне, але за його використання доводиться розплачуватися неефективним використанням пам'яті і низькою швидкістю. Програмісти, що пишуть мовою C++, можуть (і повинні) звільняти блоки пам'яті відразу після того, як вони перестали використовуватись. У програмах мовою Java блоки не звільняються до чергового виклику збирача сміття, періодичність роботи якого залежить від використовуваної реалізації віртуальної машини JVM. В додачу до великої витрати пам'яті процес "збирання сміття" вимагає додаткової процесорної потужності, яка в результаті стає недоступною програмі, і це призводить до уповільнення її роботи. Тому періодична робота збирача сміття може призводити до "заморожування" програми мовою Java на кілька секунд.

При роботі з зовнішніми програмами і пристроями, наприклад, під час введення/виведення інформації або при взаємодії з базою даних, бажано закрити файл або з'єднання з базою даних відразу ж після того, як вони перестали бути потрібні. Завдяки деструкторами C++ це відбувається відразу після виклику оператора delete. У Java закриття відбудеться лише під час наступного циклу роботи збирача сміття. У кращому випадку це може призвести до зайвого блокування ресурсів, в гіршому — до порушення цілісності відкритих ресурсів.

Головна причина, за якої процес "збирання сміття" є більш дорогим, ніж безпосереднє управління пам'яттю програмістом, — це втрата інформації. У програмі мовою C++, програміст знає і місцезнаходження своїх блоків пам'яті (зберігаючи вказівники на них), і коли вони

перестануть бути йому потрібними. У програмі, написаній мовою Java, остання інформація недоступна для віртуальної машини JVM (навіть якщо вона відома програмісту), тому JVM повинна перебирати всі блоки на предмет відсутніх вказівників. Для того, щоб викликати "збирання сміття" вручну, програміст, який пише мовою Java, може видалити всі вказівники на більше непотрібні йому блоки пам'яті. Але з боку програміста це вимагає більше зусиль, ніж безпосереднє управління пам'яттю в C++; і тим не менш, під час "збирання сміття" для JVM все одно доведеться перевірити всі блоки пам'яті, щоб звільнити невикористовувані [6].

Інструментарій Qt використовує більш ефективний підхід для спрощення завдання управління пам'яттю: при видаленні об'єкта, всі залежні від нього об'єкти також автоматично видаляються. Підхід Qt не заважає програмістам за бажанням самостійно видаляти об'єкти.

Тому при порівняній продуктивності, програмування мовою C++ забезпечує програмам швидкість роботи і ефективність використання пам'яті набагато краще, ніж при роботі з мовою Java.

Порівняння AWT/Swing і Qt

Інструментарій AWT (Abstract Windowing Toolkit) почав поставлятися з найпершою версією мови Java. Він використовує рідні для платформ компоненти GUI (тобто Win32 API для Windows і бібліотеку Motif для Unix), забезпечуючи таким чином переносну обгортку. Це означає, що зовнішній вигляд і поведінка програм, які використовують інструментарій AWT, буде відрізнятися на різних платформах, тому що саме вони займаються відображенням і управлінням компонентів GUI. Це суперечить філософії крос-платформності Java і може бути пояснено тим, що перша версія AWT була розроблена за чотирнадцять днів.

З цієї та інших причин AWT був доповнений інструментарієм Swing. Swing використовує AWT (і, отже, низькорівневі бібліотеки) тільки лише для базових операцій: створення прямокутних вікон, управління подіями і

відтворення графічних примітивів. Всім іншим, включаючи обробку компонентів GUI, займається Swing. Це вирішує проблему різниці зовнішнього вигляду і поведінки додатків на різних платформах. Але через реалізацію Swing-інструментарію засобами Java його продуктивність залишає бажати кращого. В результаті Swing-програми повільні не тільки під час інтенсивних обчислень, але і при відображенні елементів призначеного для користувача інтерфейсу.

При розробці інструментарію Qt був використаний такий же підхід: низькорівневі бібліотеки використовуються тільки для базових операцій, а відображенням елементів GUI займається безпосередньо Qt. Завдяки цьому інструментарій Qt має всі переваги Swing (наприклад, схожість поведінки і зовнішнього вигляду додатків на різних платформах), і не має проблем, пов'язаних з низькою продуктивністю, так як розроблений мовою C++ і відкомпільований у машинний код. Інтерфейс, створений за допомогою Qt, відрізняється швидкою роботою, і, завдяки використанню кешування, може бути швидше інтерфейсу, розробленого стандартними засобами. Теоретично, оптимізована не-Qt програма повинна бути швидше аналогічної Qt-програми; але на практиці для оптимізації не-Qt програми потрібно більше зусиль і майстерності, ніж для створення оптимізованої Qt-програми [7].

І Qt, і Swing підтримують техніку стилів, яка дозволяє програмам незалежно від платформи використовувати один із стилів інтерфейсу. Це стає можливим завдяки тому, що відображенням елементів GUI займаються безпосередньо Qt і Swing. Разом з Qt поставляються стилі, які емулюють зовнішній вигляд Win32, Motif, MacOS X Aqua (в Macintosh-версії), і навіть стиль, що емулює зовнішній вигляд програм Swing.

Парадигми програмування в Qt і Swing

Незважаючи на те, що оцінка API певною мірою є справою особистих уподобань програміста, серед API-інтерфейсів можна виділити

такі, які зроблять розроблений код більш простим, коротким і зрозумілим, ніж інші. Нижче наведено два приклади коду: перший з використанням Java/Swing (рис. 2.1.1), а другий з використанням C++/Qt (рис. 2.1.2), в яких реалізується вставка декількох елементів в ієрархічне дерево.

```
  ...
DefaultMutableTreeNode root = new DefaultMutableTreeNode( "Root" );
DefaultMutableTreeNode child1 = new DefaultMutableTreeNode( "Child 1" );
DefaultMutableTreeNode child2 = new DefaultMutableTreeNode( "Child 2" );
DefaultTreeModel model = new DefaultTreeModel( root );
JTree tree = new JTree( model );
model.insertNodeInto( child1, root, 0 );
model.insertNodeInto( child2, root, 1 );
...
```

Рис. 2.1.1. Приклад коду з використанням Swing

```
  ...
QListView* tree = new QListView;
QListViewItem* root = new QListViewItem( tree, "Root" );
QListViewItem* child1 = new QListViewItem( root, "Child 1" );
QListViewItem* child2 = new QListViewItem( root, "Child 2" );
...
```

Рис. 2.1.2. Приклад коду з використанням Qt

Як видно з рисунків, бібліотека Swing використовує архітектуру Model-View-Controller (MVC), в той час як бібліотека Qt її підтримує, але не нав'язує використовувати. Тому код з використанням Qt більш інтуїтивний.

Другим цікавим моментом є те, як різні інструментарії пов'язують взаємодію користувача з певною функціональністю (викликом функції або методу). Наприклад, вибір елемента у вище створеному дереві (рис. 2.1.3 та 2.1.4). Синтаксично в Swing і Qt це виглядає по-різному, але основний принцип загальний.

Важко сказати, який код є більш простим і зрозумілим. З одного боку, код з використанням Swing виглядає простіше, а з іншого - код з

використанням Qt більш гнучкий. Qt дозволяє програмісту використовувати для керуючої функції будь-яке ім'я, в той час, як Swing зобов'язує використовувати в якості імені valueChanged() (ось чому в наведеному вище прикладі з Swing воно не було вказано явно). Також Qt дозволяє пов'язувати подію ("сигнал" в термінології Qt) з будь-яким числом керуючих функцій ("слотів").

```
  ...
tree.addTreeSelectionListener( handler );
...
```

Рис 2.1.3. Приклад коду вибору елементу в дереві з використанням Swing

```
  ...
connect( tree, SIGNAL( itemSelected( QListViewItem* ) ),
         handler, SLOT( handlerMethod( QListViewItem* ) ) );
...
```

Рис 2.1.4. Приклад коду вибору елементу в дереві з використанням Qt

Таким чином, і AWT/Swing, і Qt однаково добре підходять для розробки складного користувальницького інтерфейсу. Головним недоліком інтерфейсу з використанням Swing є низька продуктивність Java.

Що стосується порівняння GUI-бібліотек, Swing і Qt, то видно, що більш нижча продуктивність програм Java робить платформу Java/Swing менш придатною для розробки GUI-додатків, навіть при однаковому досвіді програмування. Такий же самий висновок можна зробити про .NET з мовою C#.

Тому для розробки системи автоматизації розрахунку харчування на борту, було обрано мову програмування C++ з використанням платформи Qt.

2.2 Бібліотека Qt

Qt — (варіант вимови від розробників cute — к'ют) — крос-платформна бібліотека розробки програмного забезпечення мовою програмування C++ (і не тільки), яка широко використовується для створення графічних інтерфейсів.

З використанням бібліотеки Qt написано багато популярних програм: Kaspersky Internet Security, Virtual Box, Skype, VLC Media Player, Opera та інші. KDE — одне з середовищ робочого столу із багатьох програм для Linux, яке також написане із застосуванням Qt.

Qt повністю об'єктно-орієнтована, крос-платформна. Написаний код можна компілювати для Linux, Windows, Mac OS X і інших операційних систем. Функції бібліотеки Qt написані таким чином, що при компіляції під різні платформи вони "підлаштовуються" до особливостей створення графічних елементів конкретної операційної системи. Таким чином, для перенесення програми, написаної за допомогою Qt, на нову операційну систему, достатньо лише скомпілювати певним чином написаний код, не вносячи в нього поправки [8].

Бібліотека Qt містить всі основні класи, які потрібні для розробки прикладного програмного забезпечення, починаючи з класів для створення елементів графічного інтерфейсу й закінчуючи класами для роботи з мережею, базами даних, XML та ін.

Бібліотека Qt використовує систему MOC (Meta Object Compiler) для попередньої компіляції програм. Оригінальний текст програми обробляється системою MOC, яка шукає в класах програми макрос Q_OBJECT і переводить вихідний код в мета-об'єктний код, після чого мета-об'єктний код компілюється компілятором мови C++. Система MOC розширює функціональність фреймворку, завдяки їй додаються такі поняття, як слоти і сигнали.

У бібліотеці Qt є величезний набір віджетів (Widget), таких як: кнопки, прогрес бари, перемикачі, чекбокси і інші - вони забезпечують стандартну функціональність графічного інтерфейсу користувача.

Як видно на рис. 2.2.1, програми мають різний зовнішній вигляд в різних операційних системах.

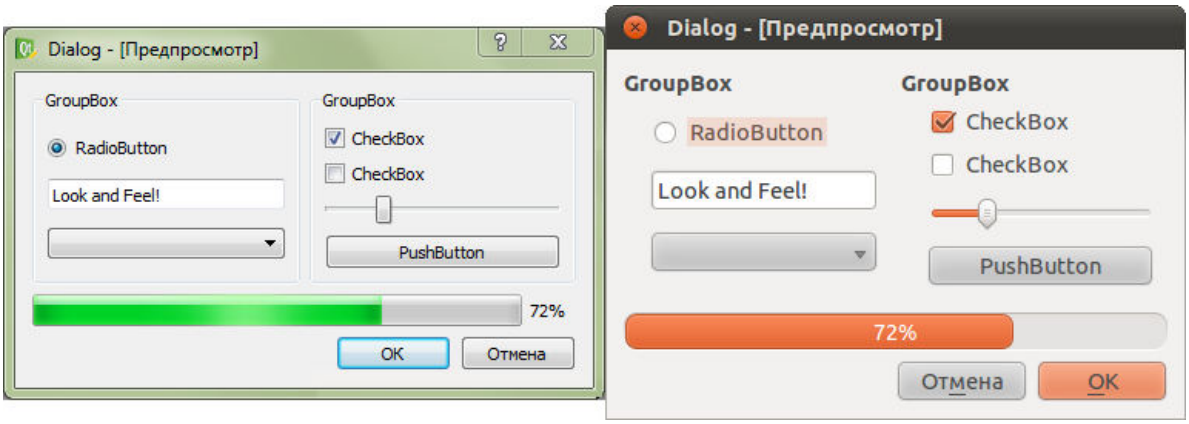

Рис. 2.2.1. Одне вікно в різних операційних системах

Для бібліотеки Qt створено середовище розробки Qt Creator. Воно включає в себе інструментарій Qt Designer, за допомогою якого можна створювати графічний інтерфейс. Візуальне створення інтерфейсу дозволяє легко і просто створювати інтерфейс, перетягуючи різні віджети (випадаючі списки, кнопки, перемикачі) на форму користувацького інтерфейсу. На рис. 2.2.2 зображено вікно редагування форми користувацького інтерфейсу в інструментарії Qt Designer [9].

Бібліотека Qt поставляється разом з Qt Assistant — це інтерактивний довідник, що містить інформацію для роботи з бібліотекою Qt. До складу Qt також входить бібліотека Qt Linguist, яка дозволяє локалізувати додаток для різних мов.

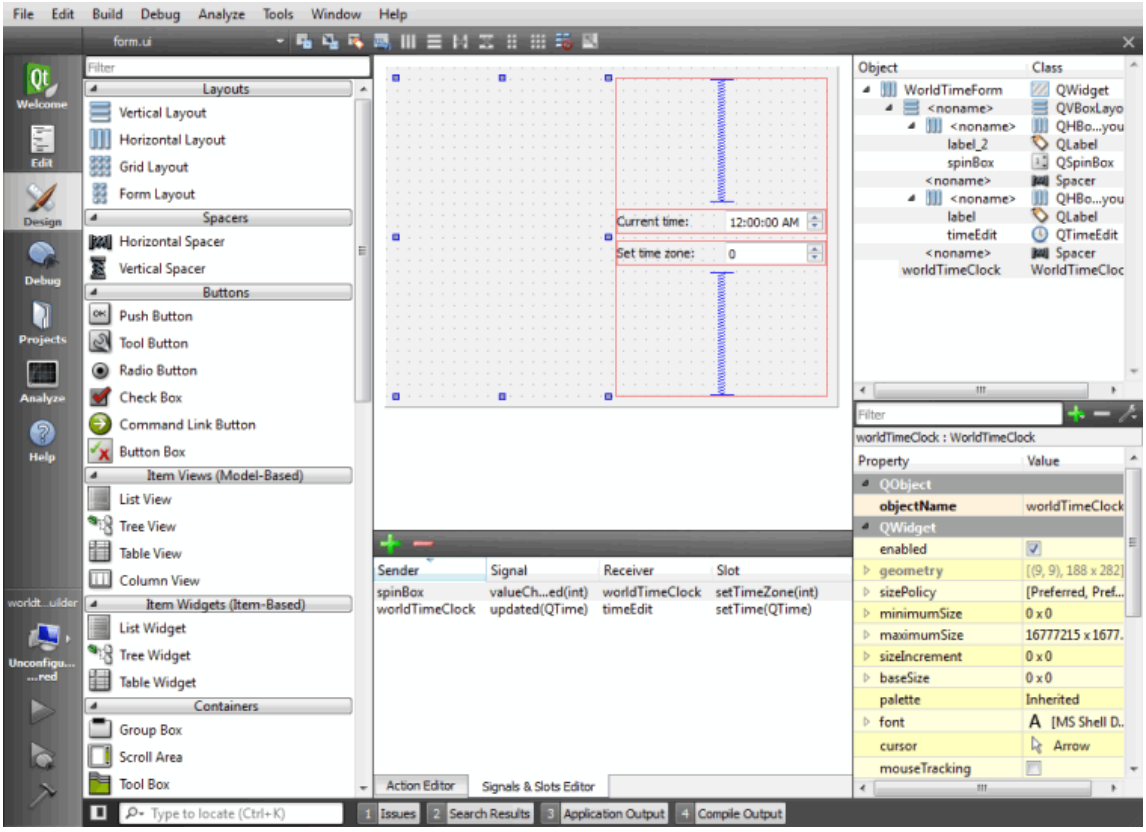

Рис. 2.2.2. Вікно редагування форми в Qt Designer

Склад бібліотеки Qt

Бібліотека Qt складається з різних модулів, які підключаються за допомогою директиви #include.

До складу бібліотеки Qt входять такі модулі:

- QtCore — модуль, що містить класи ядра бібліотеки Qt, вони використовуються іншими модулями;
- QtGui — модуль, що містить компоненти графічного інтерфейсу;
- QtNetwork — модуль, що містить класи для роботи з мережею. У нього входять класи для роботи з протоколами FTP, HTPP, IP та іншими;
- QtOpenGL — модуль, що містить класи для роботи з OpenGL;
- QtSql — модуль, що містить класи для роботи з різними базами даних з використанням мови SQL;

- QtSvg — модуль, що містить класи, що дозволяють працювати з даними Scalable Vector Graphics;
- QtXml — модуль, що містить класи для роботи з XML;
- наявні і інші модулі.

На даний момент бібліотека Qt поширюється за 3-ма ліцензіями: Qt Commercial (комерційна), GNU GPL, GNU LGPL.

Сьогодні Qt фреймворк активно розвивається. Має інтуїтивно зрозуміле API, документацію з великою кількістю прикладів, потужне середовище розробки QtCreator і широкий вибір додаткового інструментарію [10].

У той час як технології Java/AWT/Swing і C#/Micorsoft .NET Framework забезпечують розробникам дуже велику кількість бібліотек, технологія C++/Qt забезпечує програмам більшу швидкість роботи програм і кращу ефективність використання пам'яті. До того ж програми написані з використанням C++/Qt мають кращу портативність, оскільки для перенесення програми на іншу операційну систему, достатньо лише попередньо скомпілювати код, під дану платформу, не вносячи зміни у код, на відміну від технологій Java/AWT/Swing і C#/Micorsoft .NET Framework, які вимагають встановлення середовища (віртуальної машини) для виконання програми.

У даному розділі було розглянуто три технології: мова програмування C++ з бібліотекою Qt, мова Java з AWT/Swing і C# з платформою Microsoft .NET Framework, надано оцінку щодо придатності їх використання для ефективної розробки високопродуктивних програм з графічним інтерфейсом користувача.

3. РОЗРОБКА СИСТЕМИ АВТОМАТИЗАЦІЇ РОЗРАХУНКУ ХАРЧУВАННЯ НА БОРТУ

3.1 Загальна структура системи

Для розробки системи в дипломному проекті використовується архітектурний шаблон проектування модель-вигляд-контролер (Model-View-Controller, MVC). Даний шаблон передбачає поділ системи на три взаємопов'язані частини: модель даних, вигляд (інтерфейс користувача) та контролер (модуль керування). Шаблон MVC Застосовується для відокремлення даних (моделі) від інтерфейсу користувача (вигляду) так, щоб зміни інтерфейсу користувача мінімально впливали на роботу з даними, а зміни в моделі даних могли здійснюватися без змін інтерфейсу користувача [12]. Мета шаблону — гнучкий дизайн програмного забезпечення, який повинен полегшувати подальші зміни чи розширення програм, а також надавати можливість повторного використання окремих компонентів програми. Крім того, використання цього шаблону у великих системах сприяє впорядкованості структури і робить їх більш зрозумілими за рахунок зменшення складності. На рис. 3.1.1 зображена діаграма взаємодії між трьома компонентами шаблону.

Рис. 3.1.1 Діаграма взаємодії між компонентами шаблону MVC

Якщо об'єднати об'єкти вигляду і контролера, то в результаті отримуємо архітектуру модель-вигляд (див. рисунок 3.1.2). Така архітектура все ще відділяє спосіб збереження даних від способу їх представлення користувачу, але забезпечує просту структуру, засновану на тих же принципах.

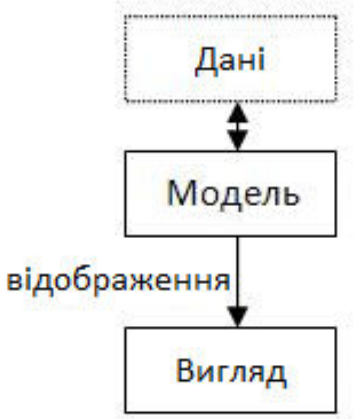

Рис. 3.1.2. Архітектура модель-вигляд

Модель і вигляд взаємодіють між собою, використовуючи сигнали і слоти: сигнали від моделі інформують вигляд про зміну даних в джерелі даних, сигнали від вигляду надають інформацію про взаємодію користувача з відображеними елементами, слоти у обох компонентів обробляють прийняті сигнали [11].

Розроблена система складається з 6 модулів, кожен з яких має власне функціональне призначення. На рис. 3.1.3 зображена схема архітектури розроблюваної системи.

Моделлю системи є головний модуль, який взаємодіє з виглядом — графічним інтерфейсом.

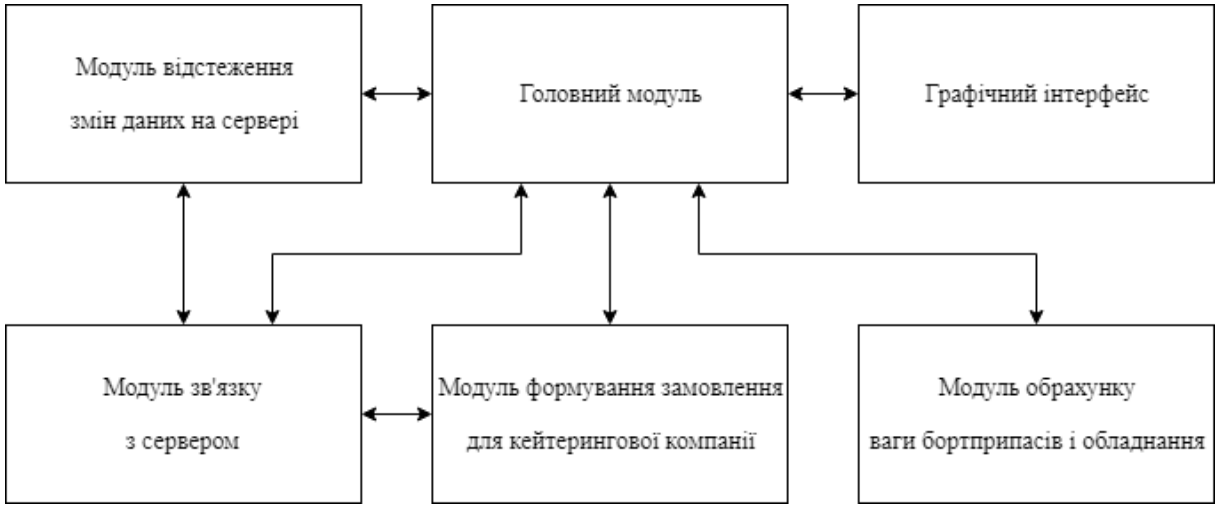

Рис. 3.1.3. Структурна схема архітектури розроблюваної системи

Призначення модулів системи:

- графічний інтерфейс — відображає дані, отримані з головного модуля, користувачу та графічні елементи для взаємодії користувача з головним модулем;
- головний модуль — надає необхідний набір функцій для графічного інтерфейсу і є посередником між графічним інтерфейсом та іншими модулями системи;
- модуль зв'язку з сервером — надає інтерфейс для зв'язку з веб-сервером;
- модуль відстеження змін даних на сервері — зчитує дані з серверу та сповіщає головний модуль про їх зміну, а той у свою чергу — графічний інтерфейс;
- модуль формування замовлення для кейтерингової компанії — здійснює всі необхідні операції для формування файлу замовлення для кейтерингової компанії;
- модуль обрахунку бортприпасів і обладнання — обчислює вагу бортприпасів і обладнання на кожному з рейсів повітряного судна в

залежності від отриманих параметрів, пов'язаних з організацією харчування на борту ПС.

3.2 Головний модуль

Головний модуль зв’язує графічний інтерфейс користувача з іншими модулями системи: приймає вхідні дані від інтерфейсу користувача, викликає необхідні функції інших модулів, передаючи ці дані як аргументи, повертає результат обрахунків графічному інтерфейсу для відображення, відслідковує зміну даних у модулях та повідомляє графічний інтерфейс про зміну.

3.3 Модуль зв'язку з сервером

Працівник відділу харчування авіакомпанії замовника використовує веб-сайт авіакомпанії для отримання інформації про кількість пасажирів на рейсах повітряних суден та замовлені пасажирами меню. Для отримання такого виду інформації працівники проходять авторизацію на веб-сайті.

Будь-який веб-сайт обслуговується веб-сервером — програмним забезпеченням, яке доставляє веб-зміст клієнтам за веб-протоколами. У розроблюваній системі наявний модуль зв'язку з сервером, що забезпечує взаємодію системи з веб-сервером, за допомогою протоколу HTTPS [12].

Розроблений модуль зв'язку з веб-сервером сайту має функціонал для посилання запиту на авторизацію та запиту на отримання інформації про усі рейси ПС за заданий день.

Для реалізації використовується бібліотека Qt Network, що містить класи з інтерфейсами для передачі даних протоколом HTTPS. Дана бібліотека самостійно виконує шифрування та використання цифрових сертифікатів, необхідних для TLS технології.

Кожне повідомлення протоколу HTTP складається з 3 частин, що передаються у вказаному порядку:

1) стартовий рядок — визначає тип (метод) повідомлення;

2) заголовки — характеризують тіло повідомлення, параметри передачі і т.д.;

3) тіло повідомлення — дані повідомлення.

Протокол HTTP має багато методів запиту інформації, проте у розроблюваній системі використовується лише один — POST. Даний метод HTTP запиту необхідний для передачі даних користувача заданому ресурсу.

У розроблюваній системі використовується QNetworkAccessManager клас, що дозволяє системі відправляти мережеві запити і приймати відповіді, які підтримуються протоколом HTTP. Для посилання запиту на ресурс використовується метод класу QNetworkAccessManager:

QNetworkReply *post(const QNetworkRequest &request, const QByteArray &data).

Параметр request — посилання на об'єкт класу QNetworkRequest, що визначає інформацію, необхідну для відправлення запиту мережею; data — вміст повідомлення, що є масивом байтів даних. Функція повертає вказівник на новостворений об'єкт класу QNetworkReply, що буде містити відповідь відправлену сервером. Для визначення моменту, коли прийде відповідь, використовується сигнал QNetworkReply::finished(), що дозволяє зчитувати відповідь.

Для аналізу вхідних даних, що потрібні для посилання та отримання HTTP запитів, було використано браузер Google Chrome та його інструменти розробника (Chrome DevTools). На рисунку 3.3.1 зображено вікно інструментів розробника.

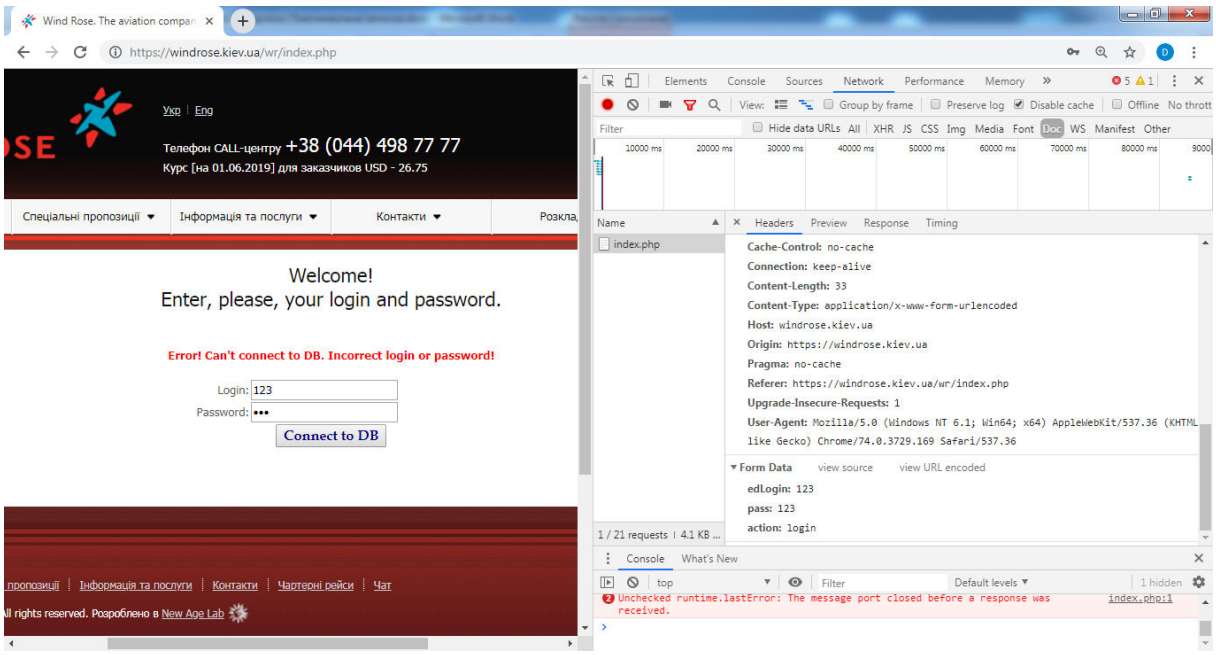

Рис. 3.3.1. Вікно інструментів розробника браузеру Google Chrome

На рис. 3.3.2 відображено алгоритм авторизації на веб-сервері, що реалізований у даному модулі.

Дані для авторизації посилаються на адресу:

https://windrose.kiev.ua/wr/index.php

При цьому тип тіла запиту вказується у заголовку Content-Type. У даному випадку вказується: application/x-www-form-urlencoded, який означає, що ключі та значення кодуються в кортежах ключ-значення, розділених символом '&' з '=' між ключем і значенням. Тому дані запиту мають вигляд:

edLogin=[логін]&pass=[пароль]&action=login

Логін і пароль отримуються з графічного інтерфейсу та вводяться користувачем.

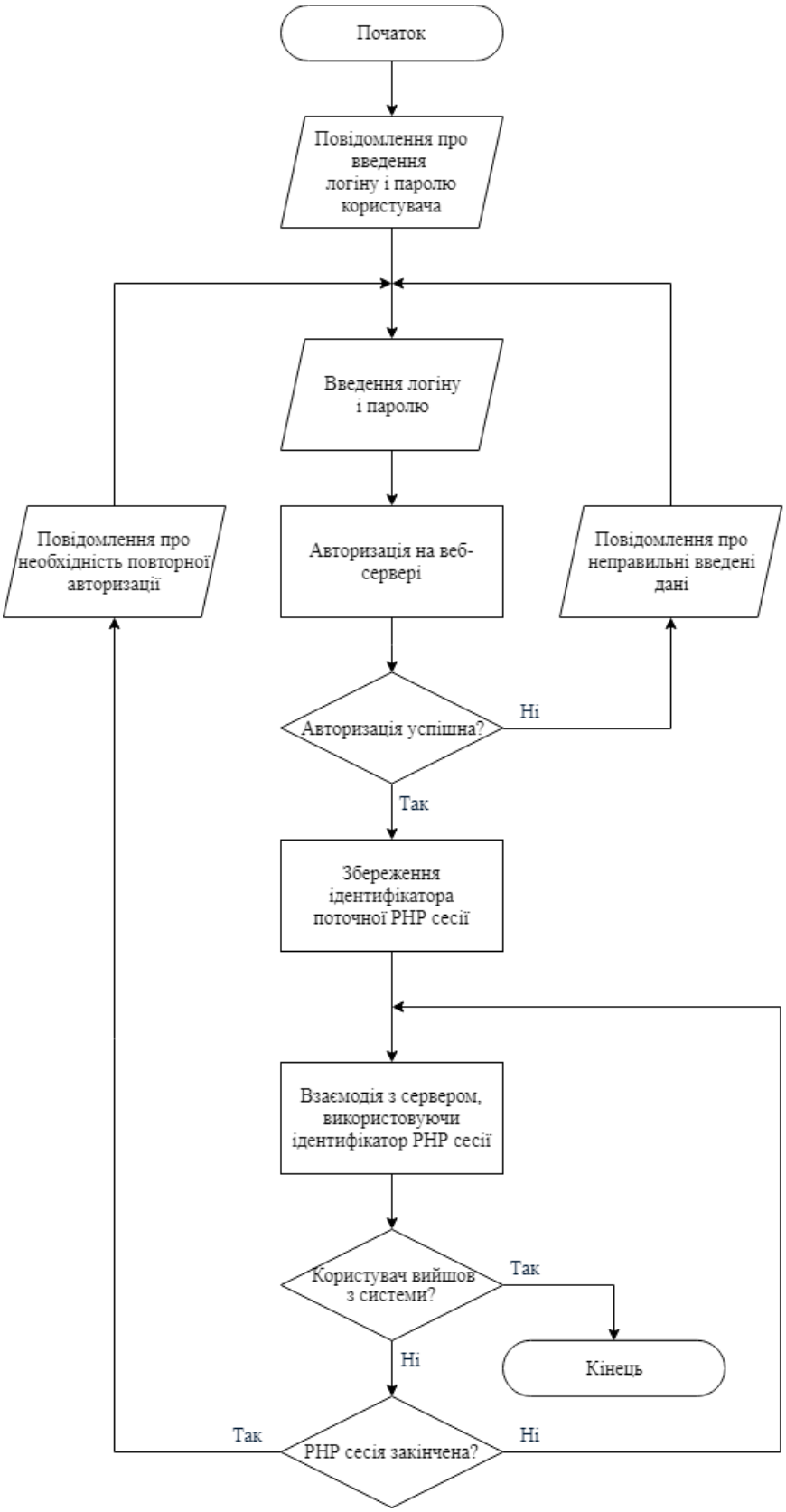

Рис. 3.3.2. Алгоритм авторизації на веб-сервері

При отримані відповіді, аналізується код стану HTTP (HTTP status code) — частина першого рядка відповіді сервера, що являє собою ціле число та відокремлена пробілом, пояснювальна фраза англійською мовою, яка вказує на причину саме такої відповіді. Код стану позначає результат запиту. Також аналізуються заголовки отриманого повідомлення. Тому для успішної авторизації необхідно, щоб код стану дорівнював — 302 Found, який означає, що подальші запити клієнта необхідно робити за адресою, вказаною у заголовку Location. Адреса, яка вказана у заголовку Location, містить ідентифікатор PHP сесії, що має вигляд: PHPSESSID=[ідентифікатор], наприклад:

PHPSESSID=0m478vsc5p2n4kaiqge5ba83p1

Сесії (сеанси) в PHP призначені для збереження відомостей про користувача при переходах між декількома сторінками. Під час сесії дані зберігаються у тимчасових файлах на сервері. Протокол HTTP являється протоколом "без збереження стану". Це означає, що даний протокол не має вбудованого способу збереження стану між двома запитами. Тобто, коли користувач відкриває спочатку одну сторінку сайту, а потім переходить на іншу сторінку того ж сайту, то ґрунтуючись тільки на засобах, які надаються протоколом HTTP, неможливо встановити, що обидва запити належать одному користувачу [13].

Для отримання інформації про кількість пасажирів на рейсах повітряних суден та замовлені пасажирами меню на заданий день, відправляється запит на адресу:

https://windrose.kiev.ua/wr/chr_FlightsListAJAX.php

У заголовок Content Type записується application/octet-stream, який означає, що будуть передані звичайні двійкові дані. Тіло запиту містить ідентифікатор поточної PHP сесії і дата у форматі рік-місяць-число. Наприклад, як виглядатимуть дані запиту інформації про рейси за 23.04.2019:

PHPSESSID=0m478vsc5p2n4kaiqge5ba83p1&DepDate=2019-04-23

Після отримання відповіді аналізується код стану та заголовок отриманого повідомлення. Якщо код дорівнює 200 OK, то у тілі повідомлення знаходиться таблиця з необхідною інформацією, яка написана мовою HTML. Якщо код дорівнює 302 Found і у заголовку Location знаходиться index.php, то це означає, що час PHP сесії вийшов та необхідна повторна авторизація для продовження роботи.

3.4 Модуль відстеження змін даних

Для відстеження змін даних на сервері використовується функціонал модулю зв’язку з сервером, що дозволяє отримати інформацію про рейси ПС за певну дату.

Алгоритм відстеження змін даних на сервері зображений на рисунку 3.4.1.

Модуль відстеження змін даних посимвольно порівнює отриману інформацію зі збереженою інформацією за заданий день. У разі зміни даних повідомлення надходить на графічний інтерфейс. В свою чергу графічний інтерфейс відображає повідомлення користувачу і подає звуковий сигнал. Аналізується інформація про рейси наступних трьох днів, при цьому поточний день є першим днем. При зміні дати поточного дня, відбувається “перезбереження” інформації: інформація про другий та третій день, записується як інформація про перший та другий день, для третього дня інформація отримується із серверу. Схему “перезбереження” можна побачити на рисунку 3.4.2.

Для періодичної перевірки змін даних використовується таймер з інтервалом часу 5 хв. Після проходження п'яти хвилин виконується перевірка зміни даних на сервері.

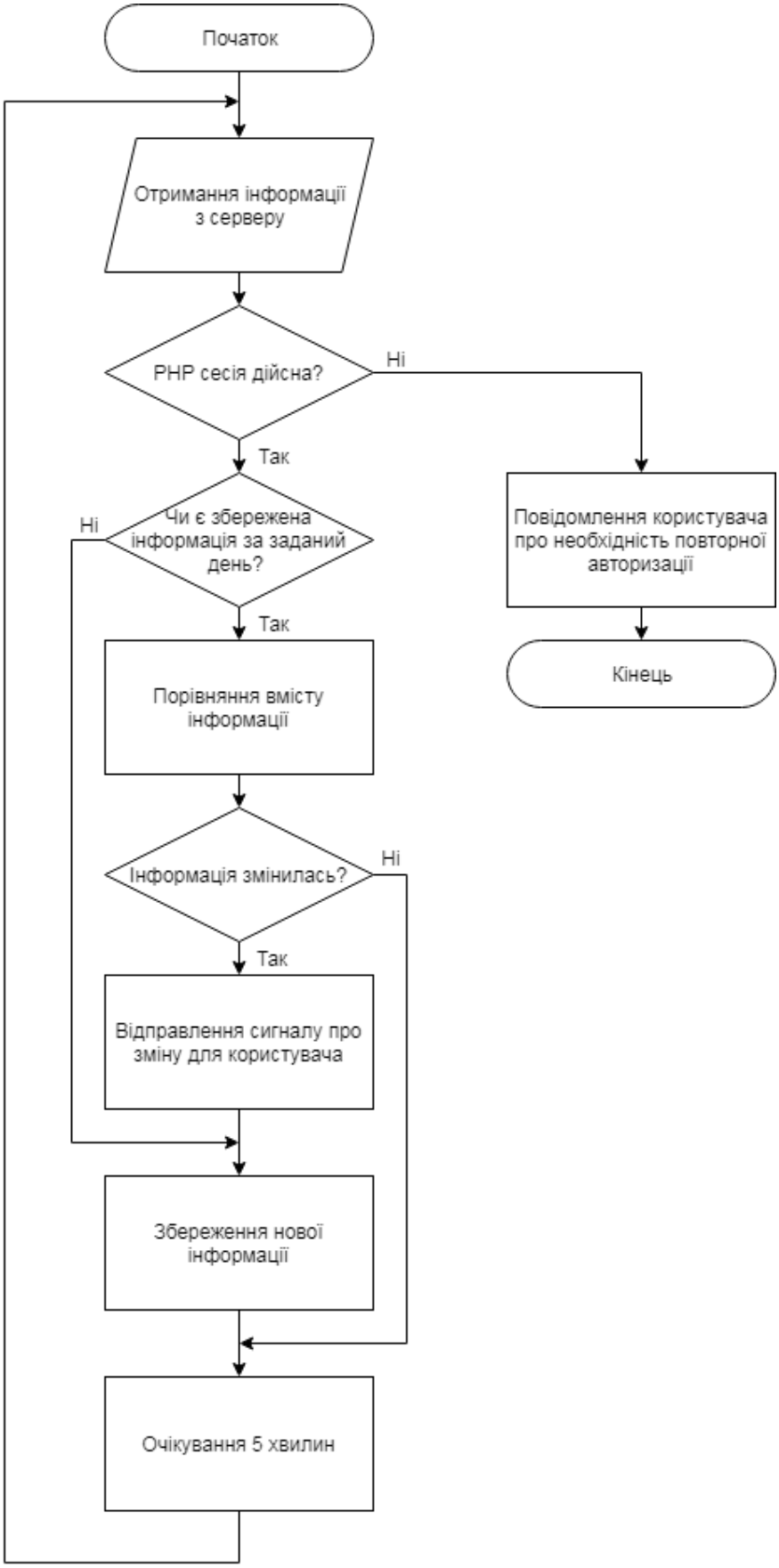

Рис. 3.4.1. Алгоритм відстеження змін даних на сервері

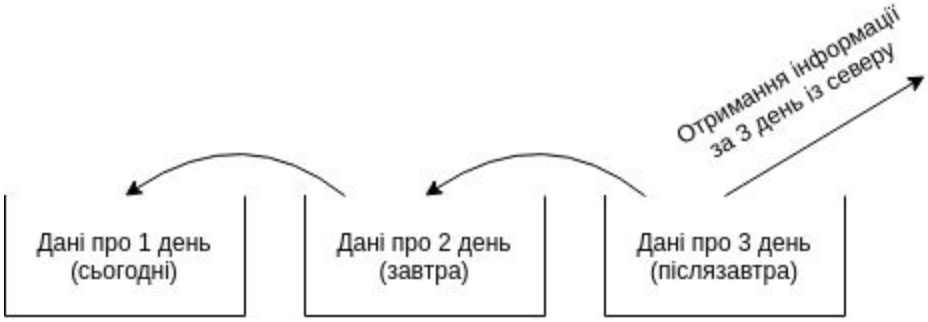

Рис. 3.4.2 Схема формування даних при зміні дати поточного дня

3.5 Модуль формування замовлення

Модуль формування замовлення надає функціонал для зчитування і обробки файлів з маршрутами повітряних суден та списком екіпажу, формує файл замовлення для кейтерингової компанії на задану дату.

На рисунку 3.5.1 відображений алгоритм формування файлу замовлення для кейтерингової компанії.

Вхідні файли з розкладом рейсів ПС і списком екіпажу, файл замовлення для кейтерингової компанії являються файлами формату xlsx. Обробка даних файлів забезпечується функціоналом бібліотеки Qt Xlsx.

Для зчитування інформації з вхідних файлів використовується функція:

QVariant read(int row, int col) const

Аргументами функції є номер рядка та стовпця файлу xlsx. Функція повертає дані типу QVariant — тип даних, який може містити об’єкт будь-якого класу даних. Даний тип являє собою обхідне рішення для мови C++, яка має строгу типізацію.

При зчитувані маршруту повітряного судна, аеропорти, через які проходить маршрут, записуються у список для подальшої обробки. Після зчитування необхідної інформації з файлу розкладу рейсів, здійснюється групування маршрутів, які утворюють цикл повітряного судна. Алгоритм групування відображений на рисунку 3.5.2.

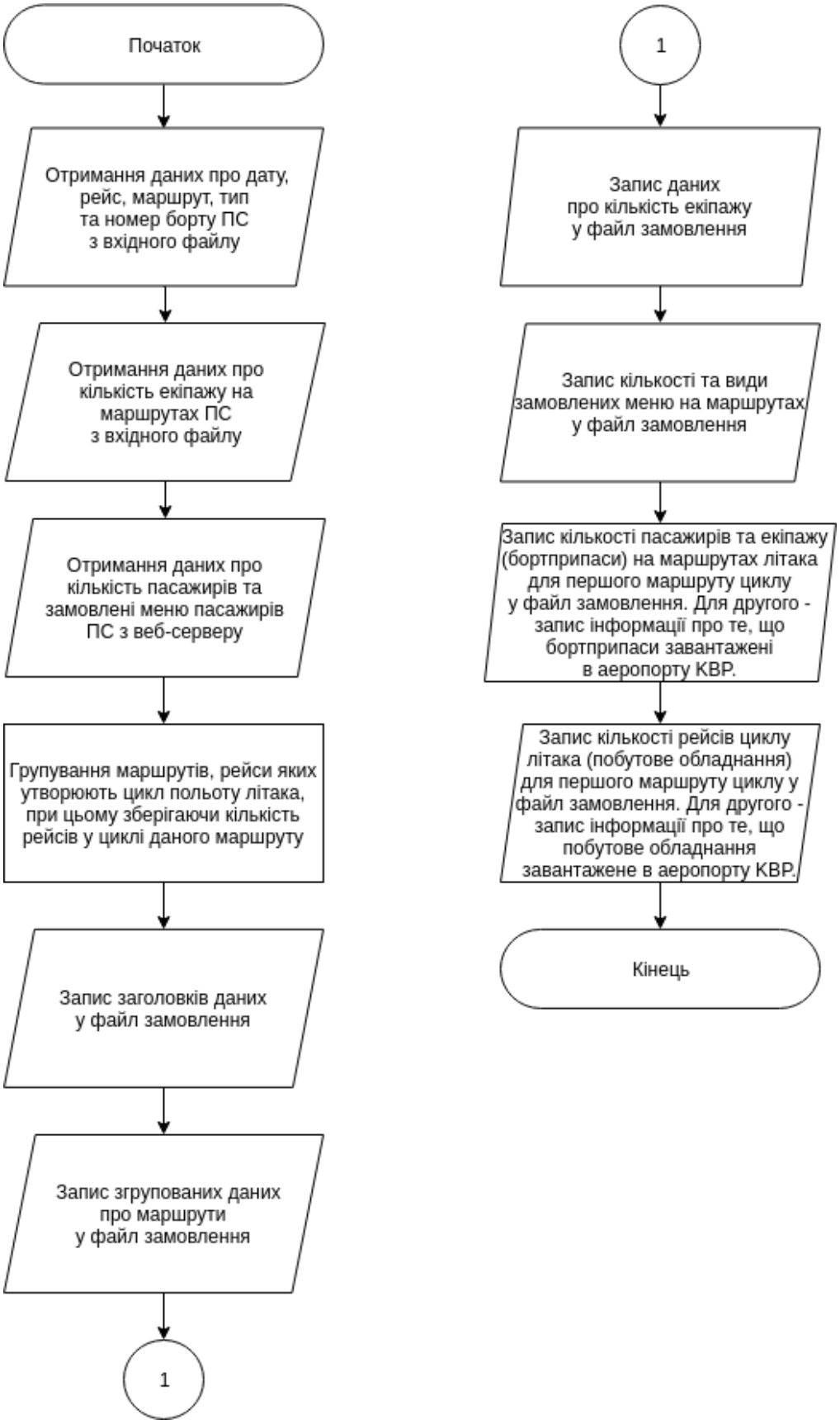

Рис. 3.5.1. Алгоритм формування файлу замовлення для кейтерингової компанії

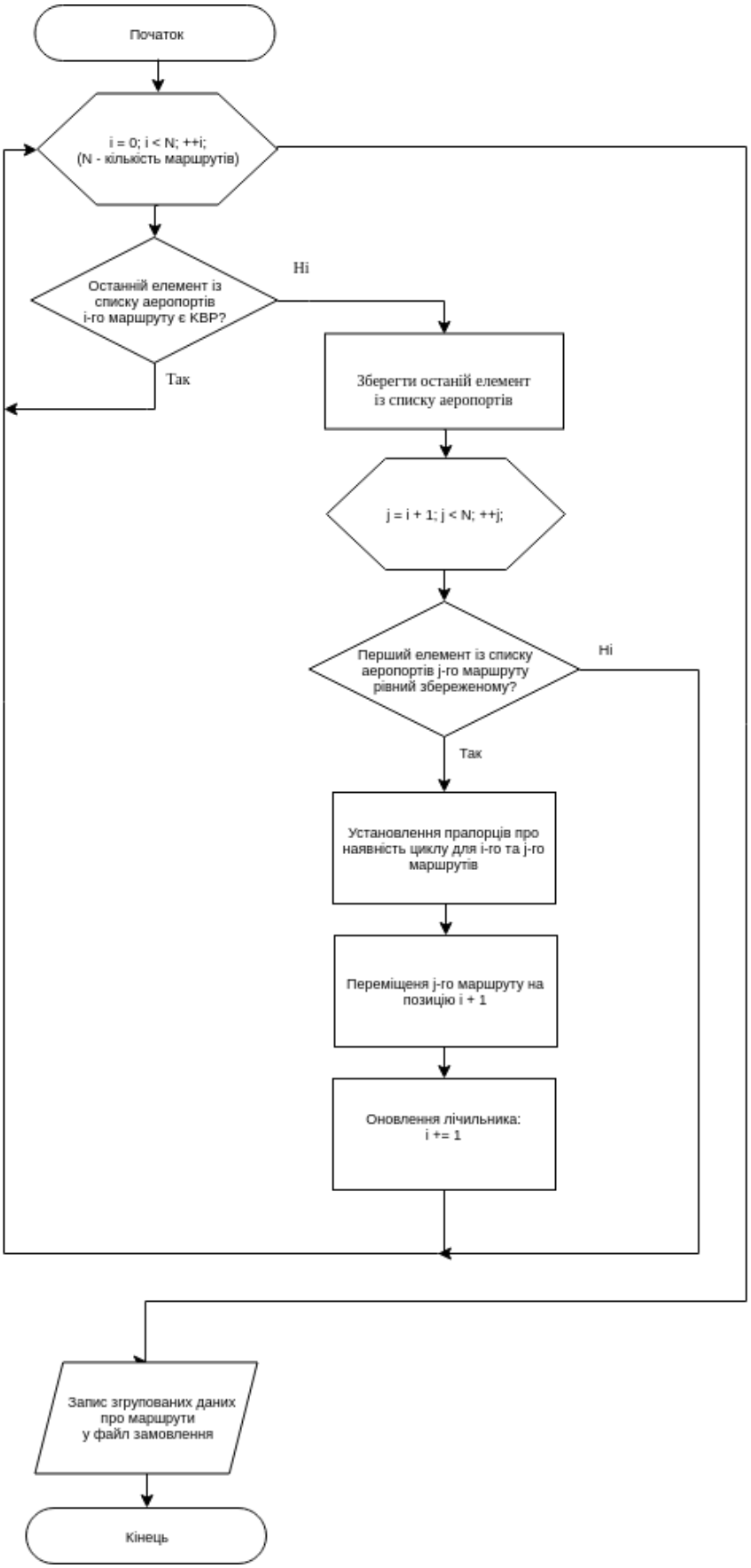

Рис. 3.5.2. Алгоритм групування маршрутів, які утворюють цикл ПС

При зчитувані списку екіпажу, враховується кількість пілотів і бортпровідників (інструктор входить до числа бортпровідників), ці дані зберігаються для маршруту, який був отриманий із файлу з розкладом рейсів.

Далі отримуються дані з серверу про кількість пасажирів на кожному з рейсів повітряного судна та замовлені ними додаткові послуги. Кількість пасажирів необхідна для розрахунку бортприпасів і розраховується як сума кількості дорослих і дітей на усіх рейсах повітряного судна. Для визначення замовлених додаткових послуг, що є позиціями Sky menu, перевіряється входження коду додаткової послуги у список кодів позицій Sky menu. У таблиці 3.5.1 визначені всі можливі коди позицій Sky menu так їх транскрипції.

Таблиця 3.5.1

Коди позицій Sky menu та їх транскрипції

№ п/п	Код харчування	Транскрипція
1	**CESK**	Холодна закуска Європейський сніданок
2	**CFSK**	Холодна закуска Фруктове асорті
3	**CGSK**	Холодна закуска М'ясне асорті
4	**CHSK**	Холодна закуска Тартар з копченим лососем
5	**CISK**	Холодна закуска Лососеві роли із сиром Філадельфія
6	**CKSK**	Холодна закуска Сирне асорті
7	**CLSK**	Холодна закуска М’ясне асорті із м’якими сортами сиру
8	**CBSK**	Салат Цезар
9	**CDSK**	Салат гриль
10	**CCSK**	Салат з лососем
11	**CASK**	Салат Грецький

Продовження таблиці 3.5.1

№ п/п	Код харчування	Транскрипція
12	**CMSK**	Десерт «Віденський штрудель»
13	**CNSK**	Десерт «Тірамісу»
14	**COSK**	Десерт «Хвилі Дунаю»
15	**CPSK**	Десерт «Панна-котта з малиновим желе»
16	**CRSK**	Десерт «Панна-котта з білого шоколаду»
17	**CSSK**	Десерт «Чізкейк»
18	**CTSK**	Десерт «Крамбл»
19	**CUSK**	Десерт Макаруни
20	**SBSK**	Сендвіч «Мега-Бейгл»
21	**SCSK**	Сендвіч «Мега-бургер зі шпинату»
22	**SDSK**	Сендвіч «Мега-бургер з чорнилами каракатиці»
23	**SESK**	Сендвіч «Мега-багет зі шпинату»
24	**SFSK**	Сендвіч «Міні-хлібець а-ля «Цезар»
25	**SGSK**	Сендвіч «Міні-хлібець з червоною рибою»
26	**SHSK**	Сендвіч «Міні-хлібець з ростбіфом»
27	**SISK**	Сендвіч «Міні-хлібець з качиною грудкою»
28	**SKSK**	Сендвіч «Міні-хлібець з м’ясним асорті»
29	**ABMP**	Fast Sky Menu Сніданок/ Омлет натуральний
30	**CBMP**	FSM Сніданок/ Фрітата зі шпинатом
31	**FBMP**	FSM Сніданок/ Запіканка ванільна із сиру
32	**BLMP**	FSM Обід/ Рибний шашлик із сьомги, сайди та картопля
33	**FLMP**	FSM Обід/ Стейк із індички та пюре із броколі
34	**ELMP**	FSM Обід/ Яловичина тушкована з кус-кусом
35	**MPSK**	FSM Обід/ Куряча брізоль та паста
36	**MRSK**	FSM Обід/ Філе птиці з рисом
37	**MSSK**	FSM Обід/ Чахохбілі з картопляним пюре
38	**MTSK**	FSM Обід/ Бургер з пастою
39	**BBML**	Special sky menu харчування для дітей до 3-ох років
40	**CHML**	SSM харчування для дітей до 12-ти років

Продовження таблиці 3.5.1

№ п/п	Код харчування	Транскрипція
41	**DBML**	SSM Діабетичне харчування
42	**VGML**	SSM Вегетаріанське харчування
43	**AVML**	SSM Азійсько-вегетаріанське харчування
44	**VLML**	SSM Лакто-вегетаріанське харчування
45	**LSML**	SSM Безсольове харчування
46	**NLML**	SSM Безмолочне харчування
47	**SFML**	SSM Харчування з морепродуктами
48	**HNML**	SSM Індуське харчування "Хінді"
49	**MOML**	SSM Мусульманське харчування
50	**GFML**	SSM Безклейковинне харчування
51	**LCML**	SSM Низькокалорійне харчування
52	**KSML**	SSM Кошерне харчування
53	**MCSK**	Sky Chef Menu Сніданок/ Омлет натуральний з овочами
54	**MDSK**	Sky Chef Menu Сніданок/ Омлет скрамбл еггз
55	**MESK**	SCM Сніданок/ Солодкі панкейки
56	**MFSK**	SCM Сніданок/ Солодкі млинці
57	**MGSK**	SCM Сніданок/ Солодкі сирники для дітей до 12 років
58	**MVSK**	SCM Сніданок/ Омлет натуральний
59	**MHSK**	SCM Обід/ Риба дорадо з картоплею у куркумі
60	**MISK**	SCM Обід/ Лосось гриль з картопляним пюре із селерою
61	**MKSK**	SCM Обід/ Філе птиці на грилі з грибним різотто
62	**MMSK**	SCM Обід/ Яловичий стейк з молодою картоплею
63	**MNSK**	SCM Обід/ Медальйони із свинини з овочами
64	**MOSK**	SCM Обід/ Для дітей до 12 років
65	**MWSK**	SCM Обід/ Качина грудка з полентою
66	**DASK**	Фреш апельсиновий 0,5 л
67	**DESK**	Лимонад малина 0,5 л

Продовження таблиці 3.5.1

№ п/п	Код харчування	Транскрипція
68	**DLSK**	Лимонад лимон-м'ята 0,5 л
69	**PCSK**	SKY Flowers рози 60 см 7 шт
70	**PDSK**	SKY Flowers рози 60 см 15 шт
71	**PPSK**	Преміум SKY PARTY/ фруктовий торт 600 г + ACTI
72	**PBSK**	Преміум SKY PARTY/Торт Прага 800 г + ACTI
73	**PASK**	Преміум SKY PARTY/Торт Святковий 600 г+ ACTI

Отримані дані про маршрути записуються у файл замовлення для кейтерингової компанії. При занесені даних про необхідні бортприпаси враховується умова: якщо маршрут є першим у циклі польоту повітряного судна, то для нього записуються дані про бортприпаси; якщо маршрут є другим у циклі, то записується інформація про те, що бортприпаси завантажені в аеропорту KBP. Для замовлення необхідної кількості комплектів побутового обладнання врахується кількість рейсів на маршрутах повітряного судна. При занесені інформації про кількість комплектів побутового обладнання застосовується схожий принцип як і для бортприпасів: для першого маршруту циклу записується розрахована кількість, а для другого - записується інформація про те, що побутове обладнання завантажене в аеропорту KBP.

3.6 Модуль обрахунку ваги бортприпасів і обладнання

Модуль обрахунку ваги бортприпасів і обладнання забезпечує порейсовий обрахунок ваги усіх бортприпасів для кількості екіпажу та пасажирів, з урахуванням кількості та ваги обладнання (візків для перевезення). Дані розрахунки необхідні для того, щоб працівники

авіакомпанії могли розрахувати загальну вагу завантаження повітряного судна, необхідну кількість пального на маршрут, і т.д.

Для розрахунків користувач вказує такі дані:

- кількість рейсів, які пролітає повітряне судно;
- кількість екіпажу на кожному з рейсів ПС;
- кількість пасажирів на кожному з рейсів ПС;
- норму бортприпасів для однієї людини;
- кількість візків, необхідних для перевезення харчування;
- вага порожнього візка.

Дані поступають на вхід модуля і використовуються для розрахунку ваги харчування і обладнання на кожному рейсі повітряного судна, за формулою (3.6.1):

$$P_i = (\sum_{k=i}^{n} \text{Е}_k + \sum_{k=i}^{n} \text{П}_k) * \text{Н} + \text{В} * \text{ВП}, i = \overline{1,n} \qquad (3.6.1)$$

де P_i — вага харчування та обладнання на i-му рейсі повітряного судна (кг);

E_k - кількість екіпажу на k-му рейсі повітряного судна;

$П_k$ - кількість пасажирів на k-му рейсі повітряного судна;

H - норма бортприпасів на одну людину (г);

B - кількість візків;

$ВП$ - вага порожнього візка (кг);

n - кількість маршрутів.

3.7 Графічний інтерфейс

Для розробки графічного інтерфейсу використовувались бібліотеки Qt QML і Qt Quick. Бібліотека Qt QML надає платформу для розробки додатків з використанням мови QML (Qt Meta Language або Qt Modeling

Language) - мова, що дозволяє описувати інтерфейси користувача з точки зору їх візуальних компонентів і того, як ці компоненти взаємодіють і пов’язані один з одним. Бібліотека Qt Quick забезпечує реалізацію архітектури модель-вигляд, надає основні візуальні компоненти, керування анімацією і інші опції для створення графічних інтерфейсів.

3.8 Опис роботи системи

Після запуску програми відображається вікно авторизації (рис. 3.8.1), де користувачу необхідно ввести свій логін і пароль.

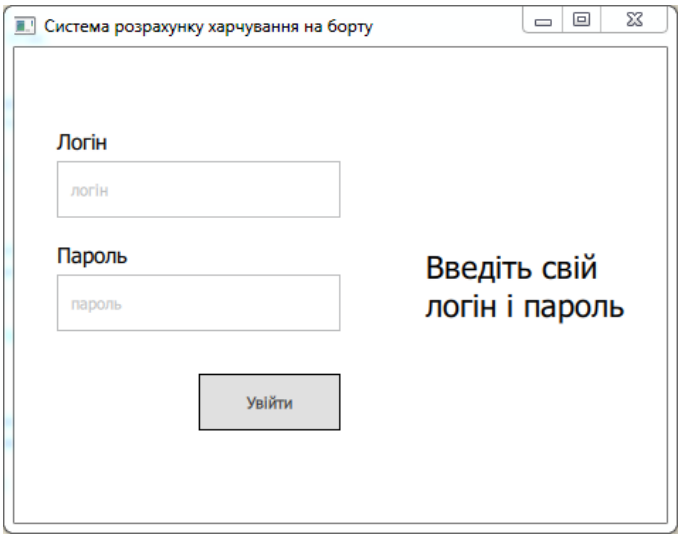

Рис. 3.8.1. Вікно авторизації

Після натиснення на кнопку "Увійти", введені дані приймаються головним модулем і передаються на веб-сервер через модуль зв'язку з веб-сервером. На веб-сервері здійснюється перевірка даних та посилається відповідь, яка приймається модулем зв'язку з веб-сервером. Результат виконання авторизації передається через головний модуль на графічний інтерфейс. Якщо введені дані правильні, то відкривається головне меню системи (рис. 3.8.2), інакше текст "Введіть свій логін і пароль" змінюється на "Неправильний логін або пароль".

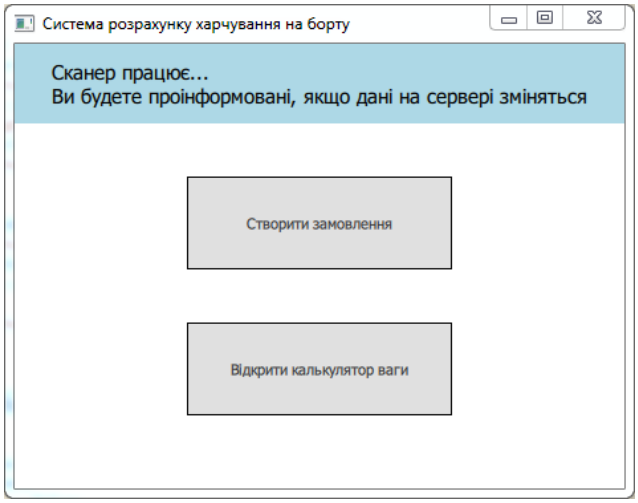

Рис. 3.8.2. Головне меню системи

Після натиснення на кнопку "Створити замовлення" відкривається меню формування замовлення (рис. 3.8.3), у якому користувач вибирає файли з розкладом усіх рейсів і списком екіпажу повітряного судна.

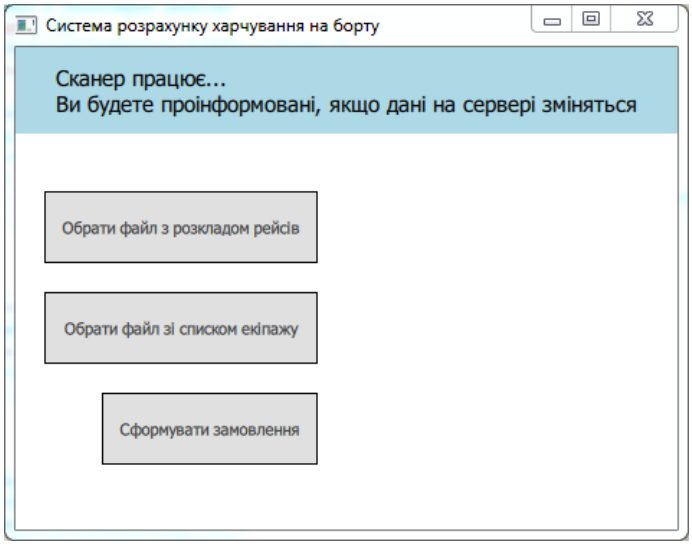

Рис. 3.8.3. Меню формування замовлення

У разі натиснення кнопки "Сформувати замовлення" без вибору необхідних файлів, відображається повідомлення про помилку (рис. 3.8.4).

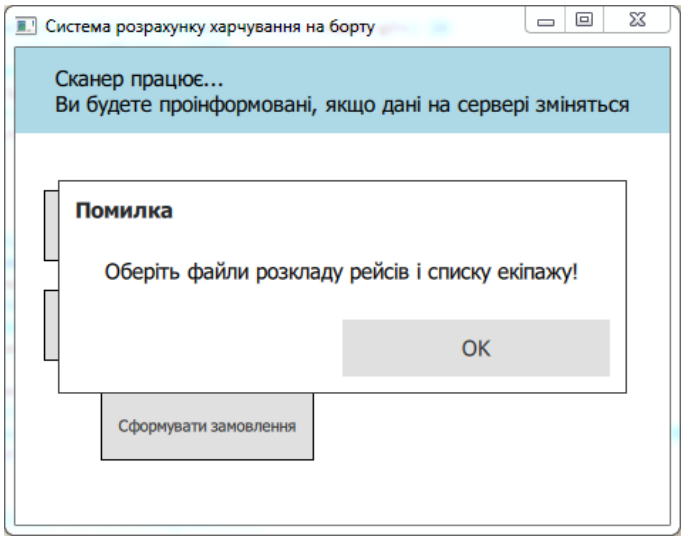

Рис. 3.8.4. Повідомлення про помилку

Після натиснення користувачем на кнопку вибору файлу у меню формування замовлення (рис. 3.8.3), відображається вікно, у якому вибирається файл визначеного формату xlsx (рис. 3.8.5).

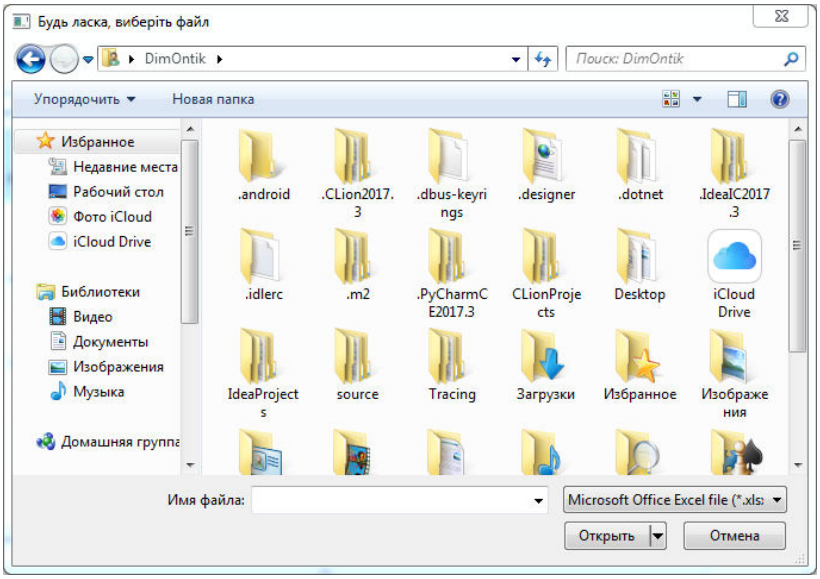

Рис. 3.8.5. Вікно вибору файлу

Обрані користувачем файли відображаються біля відповідних кнопок (рис. 3.8.6).

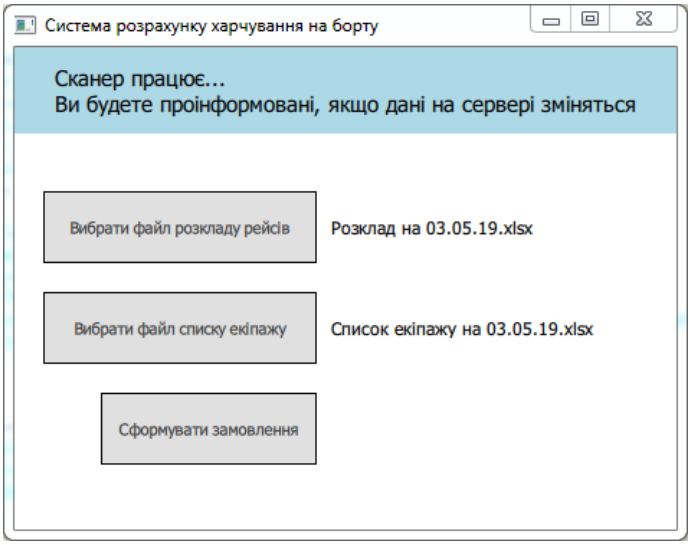

Рис. 3.8.6. Меню формування замовлення після вибору необхідних файлів

Для прикладу формування замовлення для кейтерингової компанії взято дані за 03.05.19. Вміст файлу з розкладом рейсів за 03.05.2019 відображено у таблиці 3.8.1.

Таблиця 3.8.1

Розклад рейсів повітряних суден на 03.05.2019

Примітки	Дата	Рейс	Маршрут	Тип ПС/Борт
03.05.19 П'ятниця				
	03.05.19	7W 7051/7W 2052	KBP 03:05-SSH 06:45/07:05-ODS 11:50	A-321/UR-WRV
	03.05.19	7W 7697/7W 7698	KBP 06:30-AYT 09:40/10:05-KBP 13:25	A-320/UR-WRM
	03.05.19	7W 755/7W 756	KBP 09:05-SOF 11:25/ 12:00-KBP 14:15	E145/UR-DNR
	03.05.19	7W 2051/7W 7052	ODS 12:50-SSH 15: 50/ 16:15-KBP 19:20	A-321/UR-WRV
	03.05.19	7W 759/7W 760	KBP 14:40-OTP 16:45/17:15-KBP 19:35	E145/UR-DNG

У таблиці 3.8.2 знаходиться список екіпажу для повітряних суден, які здійснюють перельоти на час заданої дати.

Таблиця 3.8.2

Список екіпажу повітряних суден за 03.05.2019

Тип ПС/Борт	Рейс	Маршрут	Виліт UTC	Екіпаж			
				Командир	2 пілот	Інструктор	Бортпровідники
UR-WRV	7W 7051 7W 2052	KBP-SSH SSH-ODS	03:05-06:45 07:05-11:50	Бабич І.А.	Макаєв І.В.		Біляшова І.В. Стах В.Л. Тюхміна О.О. Усаченко Л.В. Мартинюк В.О.
UR-WRV	7W 2051 7W 7052	ODS-SSH SSH-KBP	12:50-15:50 16:15-19:20	Волошин Г.Р.	Ларченко С.М.		Горбань К.С. Дегурко А.А. Калюжна Д. В. Конюченко О.Г. Свиридова Ю.М.
UR-WRM	7W 7697 7W 7698	KBP-AYT AYT-KBP	06:30-09:40 10:05-13:25	Стрілець В.В.	Сербін Д.М.		Малюта А.М. Осипова О.С. Перехрест Л. А. Шатохіна Т.В.
UR-DNR	7W 755 7W 756	KBP-SOF SOF-KBP	09:05-11:25 12:00-14:15	Осипов С.Т.	Грайцар Р.С.		Вдовиченко Є.Р. Загородня С. Г.
UR-DNG	7W 759 7W 760	KBP-OTP OTP-KBP	14:40-16:45 17:15-19:35	Рябой М.В.	Стецюк О.Р.		Підгорна Л.В. Третяк Н.Р. Шкуренко А.А.

Після вибору файлів користувачем, шляхи до файлів передаються через головний модуль до модуля формування замовлення для кейтерингової компанії, де відбувається обробка вхідних файлів. Далі модуль формування замовлення через модуль зв'язку з сервером отримує інформацію про кількість пасажирів та замовлені меню (табл. 3.8.3).

Вибравши файли з розкладом рейсів і списком екіпажу, користувач натискає кнопку "Сформувати замовлення", тоді з'являється повідомлення про те, що файл "Лист замовлення" для кейтерингової компанії сформований (рис. 3.8.7).

Таблиця 3.8.3

Інформація про кількість пасажирів та замовлені меню за 03.05.19

№	Рейс	Маршрут	Дата	Відправлення	Харчування	
					Код	Кількість
1	7W 7051	KBP-SSH	03.05.19	03:05	PESK	1
					PRSK	2
					Y	179
					Y Chd	41
2	7W 7697	KBP-AYT	03.05.19	06:30	FTSB	1
					PESK	1
					UMNR	1
					WCHS	1
					Y	182
					Y Chd	35
3	7W 2052	SSH-ODS	03.05.19	07:05	SPEQ	1
					Y	175
					Y Chd	43
4	7W 755	KBP-SOF	03.05.19	09:05	Y	19
5	7W 7698	AYT-KBP	03.05.19	10:05	EXST	1
					Y	110
					Y Chd	18
6	7W 756	SOF-KBP	03.05.19	12:00	Y	16
7	7W 2051	ODS-SSH	03.05.19	12:50	Y	151
					Y Chd	60
8	7W 759	KBP-OTP	03.05.19	14:40	Y	27
					Y Chd	3
9	7W 7052	SSH-KBP	03.05.19	16:15	CHML	1
					CUSK	1
					DESK	3
					ELMP	1
					MPSK	1
					MRSK	2
					PRSK	3
					Y	175
					Y Chd	45
10	7W 760	OTP-KBP	03.05.19	17:15	Y	23
					Y Chd	1

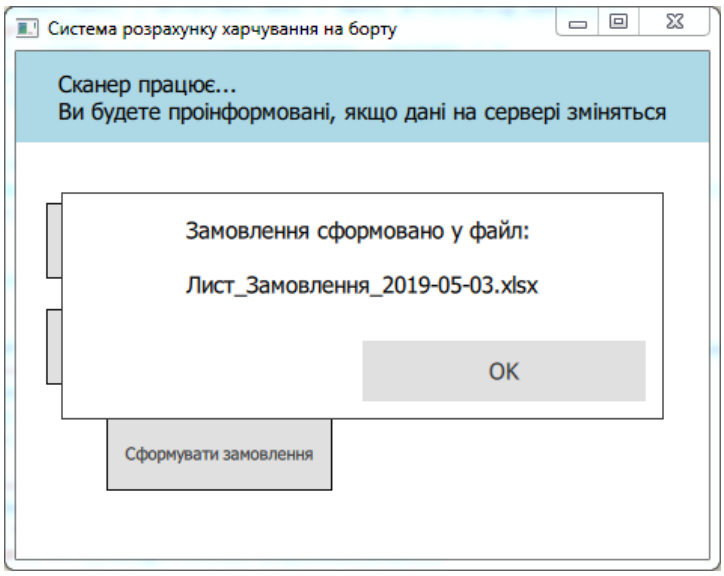

Рис. 3.8.7. Повідомлення про сформоване замовлення

Результатом розрахунку і формування замовлення є файл формату xlsx. Вигляд сформованого файлу з "Листом замовлення" за 03.05.19 відображено у таблиці 3.8.4.

У разі необхідності сповістити про зміни даних на сервері, система відображає повідомлення користувачу (рис. 3.8.8) у будь-якому вікні, окрім вікна авторизації.

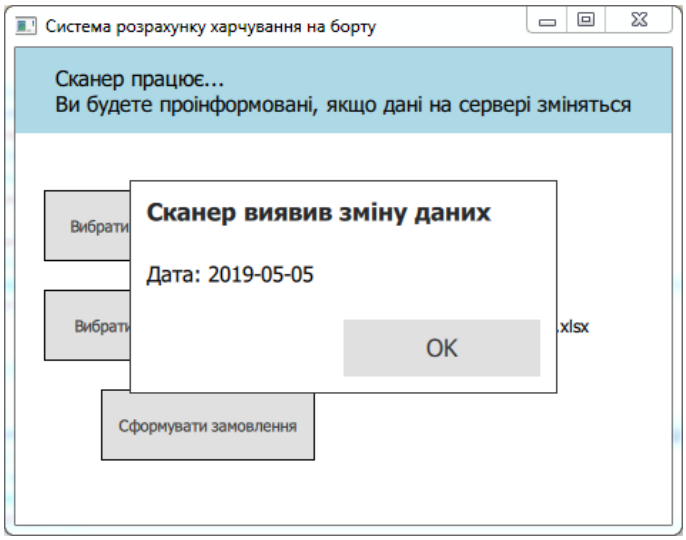

Рис. 3.8.8. Повідомлення про зміну даних на сервері

Таблиця 3.8.4

"Лист замовлення" для кейтерингової компанії за 03.05.19

Дата	№ рейсу	Тип ПС/ Бортовий	Маршрут	Пілоти		Екіпаж		Sky Food		Бізнес/ Преміум/ Sky menu		Харчування		Бортприпаси, норма	BF Group	Побутове обладнання
				Туди	Зворотнє	Туди	Зворотнє	Туди	Зворотнє	Туди	Зворотнє	Туди	Зворотнє			
03.05.19	7W 7051/ 7W 2052	A-321/ UR-WRV	KBP 03:05-SSH 06:45/ 07:05-ODS 11:50	1/1	1/1	5	5	*	*	*	*	*	*	Розрахунок бортприпасів на всі перельоти 897 Y/C	*	Побутове обладнання на 4 перельоти
03.05.19	7W 2051/ 7W 7052	A-321/ UR-WRV	ODS 12:50-SSH 15: 50/ 16:15-KBP 19:20	1/1	1/1	5	5	*	*	*	1 CHML 1 CUSK 3 DESK 1 ELMP 1 MPSK 2 MRSK	*	*	Бортприпаси з KBP	*	Побутове обладнання з KBP
03.05.19	7W 7697/ 7W 7698	A-320/ UR-WRM	KBP 06:30-AYT 09:40/ 10:05- KBP 13:25	1/1	1/1	4	4	*	*	*	*	*	*	Розрахунок бортприпасів на всі перельоти 357 Y/C	*	Побутове обладнання на 2 перельоти
03.05.19	7W 755/ 7W 756	E145/ UR-DNR	KBP 09:05-SOF 11:25/ 12:00- KBP 14:15	1/1	1/1	2	2	*	*	*	*	*	*	Розрахунок бортприпасів на всі перельоти 43 Y/C	*	Побутове обладнання на 2 перельоти
03.05.19	7W 759/ 7W 760	E145/ UR-DNG	KBP 14:40-OTP 16:45/ 17:15-KBP 19:35	1/1	1/1	3	3	*	*	*	*	*	*	Розрахунок бортприпасів на всі перельоти 64 Y/C	*	Побутове обладнання на 2 перельоти

Коли користувач натискає у головному меню (рис. 3.8.2) кнопку "Відкрити калькулятор ваги", з'являється вікно калькулятора ваги харчування і обладнання (рис. 3.8.9), де у користувача є можливість вибрати необхідну кількість рейсів, що здійснює повітряне судно та ввести вхідні дані, необхідні для розрахунку. На рисунку 3.8.9 зображений приклад із введеними даними.

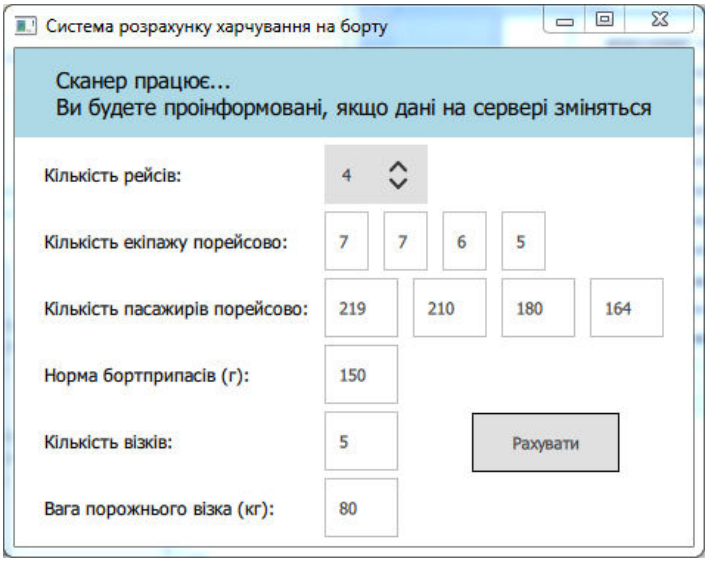

Рис. 3.8.9. Вікно калькулятора ваги харчування і обладнання

Після натиснення на кнопку "Розрахувати", відображається повідомлення з результатами обрахунків ваги харчування і обладнання на кожному з рейсів повітряного судна (рис. 3.8.10).

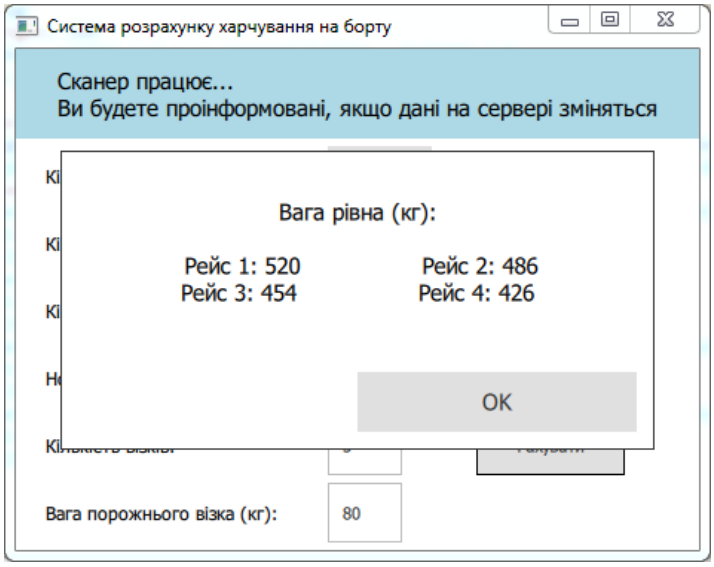

Рис. 3.8.10. Повідомлення з результатами обрахунків ваги на рейсах

При необхідності повторної авторизації, поточне відкрите вікно закривається і відображається вікно авторизації (рис. 3.8.1).

ВИСНОВКИ

Метою даного дипломного проекту було розробка системи автоматизації, яка забезпечує формування замовлень для кейтерингової компанії, що обслуговує екіпаж та пасажирів на борту повітряного судна.

Проведений аналіз існуючих рішень показав, що жодна із існуючих систем, не може бути інтегрованою до системи авіакомпанії WINDROSE, оскільки розроблена система повинна напряму взаємодіяти з системою замовника для отримання вхідної службової інформації, наприклад, опис рейсів, кількість і склад екіпажу, кількість пасажирів і т.п.

Виходячи з того, що в компанії замовника система, яка використовується працівниками компанії, що відповідають за організацію харчування на борту ПС, не забезпечує можливість повного автоматизованого формування документів для кейтерингової компанії, розробка системи автоматизації розрахунку харчування на борту є актуальною.

Для реалізації розробки системи було обрано мову програмування C++ з використанням бібліотеки Qt, оскільки технологія C++/Qt забезпечує програмам високу продуктивність роботи програм, ефективне використання пам'яті та портативність програмного продукту.

Розроблена система умовно розділена на три частини:

- перша частина отримує дані з веб-серверу авіакомпанії та відслідковує зміну інформації про рейси за певні 3 дні;
- друга частина отримує дані про маршрути літаків і склад екіпажу, здійснює необхідні обрахунки для формування замовлення на бортове харчування для кейтерингової компанії;
- третя частина розраховує вагу бортприпасів та обладнання з урахуванням введених параметрів, пов'язаних з організацією харчування на борту ПС.

Розроблена система забезпечує наступні функції:

- авторизація користувача за допомогою логіну і паролю;
- надання інформації користувачу про зміни даних на веб-сервері за певний день;
- формування замовлення для кейтерингової компанії;
- обчислення ваги бортприпасів і обладнання.

У подальшому розроблена система може бути розширена додатковим функціоналом для зручності використання.

СПИСОК ВИКОРИСТАНИХ ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Кейтеринг [Електронний ресурс]. — Режим доступу: https://uk.wikipedia.org/wiki/Кейтеринг. — Дата доступу : 15.03.2019. — Назва з екрану.
2. Губенко, А.В. Экономика воздушного транспорта [Текст] / А.В. Губенко, М.Ю. Смуров, Д.С. Черкашин — СПб. : Питер, 2014. — 288 с.
3. Осипова, О.Я. Транспортное обслуживание туристов [Текст]. — Академия, 2006. — 384 с. — ISBN 5-7695-3435-4.
4. Графічний інтерфейс [Електронний ресурс]. — Режим доступу: https://uk.wikipedia.org/wiki/Графічний_інтерфейс. — Дата доступу : 20.04.2019. — Назва з екрану.
5. Barry Boehm. Software Engineering Economics, by Barry Boehm, Prentice Hall.
6. A method of programming measurement and estimation, by C. E. Walston and C. P. Fexix, IBM.
7. An empirical comparison of C, C++, Java, Perl, Python, Rexx, and Tcl by Lutz Prechelt, University of Karlsruhe.
8. Qt [Електронний ресурс]. — Режим доступу: https://uk.wikipedia.org/wiki/Qt. — Дата доступу : 21.04.2019. — Назва з екрану.
9. Алексеев, Е.Р. Програмирование на языке C++ в среде Qt Creator [Текст] / Е.Р. Алексеев, Г.Г. Злобин, Д.А. Костюк — М. : ALT Linux, 2015. — 448 с. — ISBN 978-5-905167-16-4.
10. Шлее, М. Qt 5.10 Профессиональное программирование на C++ [Текст] — СПб. : БХВ-Петербург, 2018 — 1072 с. — ISBN 978-5-9775-3678-3.

11. Модель-вид-контролер [Електронний ресурс]. — Режим доступу: https://uk.wikipedia.org/wiki/ Модель-вид-контролер. — Дата доступу : 27.04.2019. — Назва з екрану.
12. Крол Эд. Все об INTERNET: Руководство и каталог [Текст] / Пер. с англ С. М. Тимачева. — К.: Торгово-издательское бюро BNV, 1995. — 592 с.
13. Кузнецов, М.В. Самоучитель PHP 5/6 (3-е издание). [Текст] / Кузнецов М.В., Симдянов И.В. — СПб. : БХВ-Петербург, 2009 — 672 с. — ISBN 978-5-9775-0409-6.